STSC

Software Estimation Technology Report

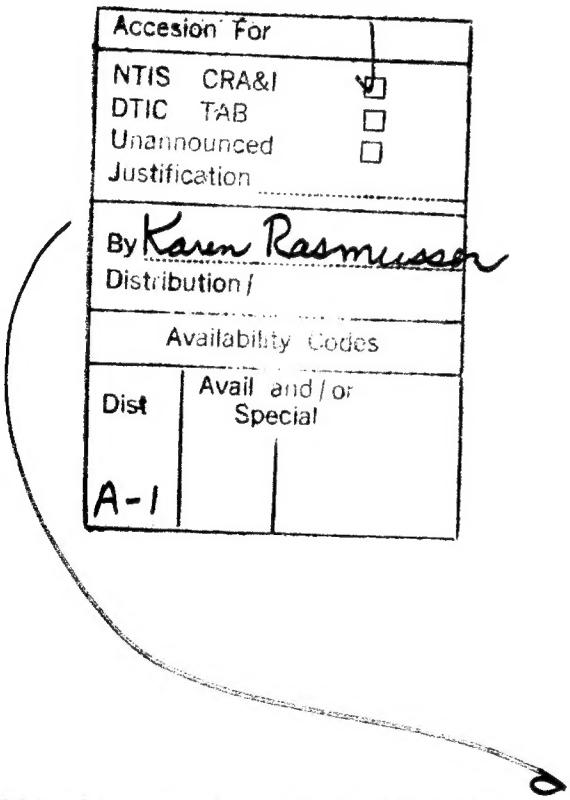
March 1993

Dean Barrow
Susan Nilson
Dawn Timberlake
Project Management Tool Evaluation Project

This technical report was prepared by the:

Software Technology Support Center
OO-ALC/TISE
Hill AFB, UT 84056

The ideas and findings in this report should not be construed as an official Air Force position. It is published in the interest of scientific and technical information exchange.



This document is available through the STSC. To obtain a copy, please contact the Software Technology Support Center, Attn: Customer Service, OO-ALC/TISE, Hill AFB, Utah 84056; 801-777-7703 or DSN 458-7703, Fax 801-777-8069.

Preface

The purpose of this report is to increase awareness and understanding of software systems estimation technology. Use of this report should be the first step in transferring effective software estimation principles, methods, and tools into practical use. The target audience of this report consists of those managers and technical people responsible for the development and maintenance of software in their organizations. This report defines the concepts of the Software Estimation Tools Domain and identifies their value in improving software quality and productivity. It explains how the capabilities of current software estimation tools can improve management of software development and maintenance projects. It includes information about specific products in the marketplace. The information is aimed at those who must make decisions about acquiring technology and who must prepare their organizations to use it effectively. Finally, this report identifies the future directions in the field to help in planning long-range strategies.

Table of Contents

List of Figures

1 INTRODUCTION

This report was written to help Air Force Software Development and Support Activities (SDSAs)[1] identify and evaluate technologies that could potentially improve their ability to estimate software development costs. Section 1 introduces the Software Technology Support Center (STSC) and its mission. Section 2 discusses software estimation. Section 3 describes the state of software estimation tools. Section 4 discusses trends in software estimation and offers conclusions.

1.1 The Software Technology Support Center

The mission of the Software Technology Support Center (STSC) is to transition technologies and exchange information to help Air Force Software Development and Support Activities continuously improve their software quality and life cycle productivity.

A planned approach is necessary for successful transition. In general, transitioning effective practices, processes, and technologies consists of a series of activities or events that occur between the time a person encounters a new idea and the daily use of that idea. Conner and Patterson's Adoption Curve [Conner 82], shown in Figure 1-1, illustrates these activities.

After encountering a new process or technology, potential customers of that technology increase their awareness of its usage, maturity, and application. If the process or technology is promising, then customers try to better understand its strengths, weaknesses, costs, and applicability. The first activities in the Adoption Curve take a significant amount of time.

Promising processes and technologies are then evaluated and compared. To reduce risk, customers usually try new processes or technologies on a limited scale through beta tests, case studies, or pilot projects. A customer then adopts processes or

---

[1] A Software Development and Support Activity is a DoD or Military Service organization responsible for the software development and/or support of a designated Mission-Critical Computer Resource (MCCR). Adaptation based on "Mission-Critical Computer Resources Software Support," *Military Handbook 347*, Department of Defense, p. 10 (May 1990).

technologies that prove effective. Finally, refined processes and technologies become essential parts of an organization's daily process (institutionalization).

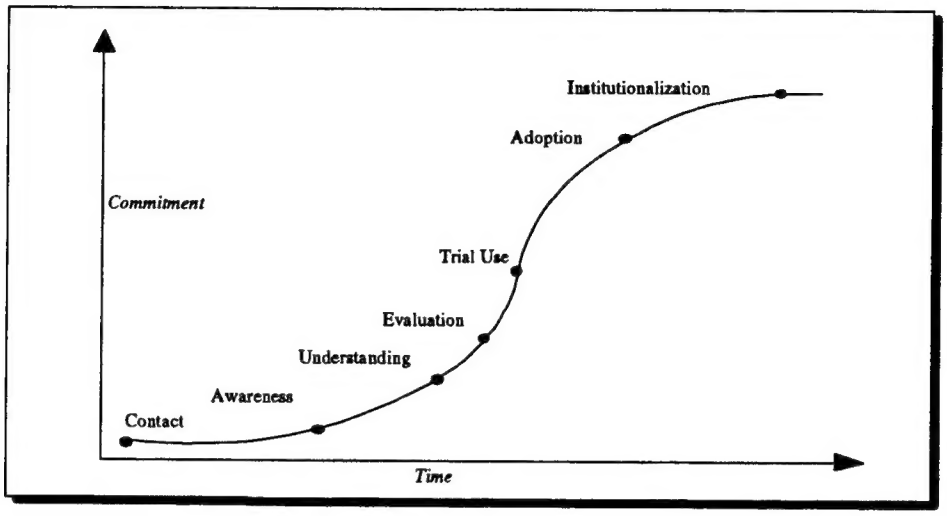


Figure 1.1 Adoption Curve

Word processors are essential in most organizations' daily operations. Yet, thirty years ago, they did not exist. The institutionalization of word processors in many organizations followed a series of events similar to those identified in the Adoption Curve.

The STSC is researching and collecting information about technologies that will reduce the time and resources it takes to become aware, understand, evaluate, test, try, and adopt effective practices, processes, and technologies. The STSC has developed the following objectives to accomplish its mission:

- Technology Evaluation:
  Identify, validate, classify, and evaluate effective processes and technologies.

- Information Exchange:
  Facilitate the exchange of better software business practices, processes, and technologies within the Air Force.

- Insertion Projects:
  Analyze and improve processes, adopt new methodologies as needed, evaluate and select effective tools, receive appropriate levels of training, and perform pilot projects to try out and confirm the technology insertion efforts.

- STSC Associates:
  Develop STSC Associates who can infuse effective process and technology improvements through the use of STSC products, services, and processes.

1.2 STSC Technology Transition Approach

This section describes the STSC's approach to meeting the objectives identified in the previous section.

1.2.1 Technology Evaluation

The first objective involves identifying, validating, and classifying processes, methods, and technologies that can potentially improve the quality or productivity of software development and maintenance. Many organizations are so focused on deadlines and customer needs that they lack the resources and time to thoroughly investigate options for improvement, leaving them vulnerable to marketing hype. The STSC has developed the infrastructure to provide information on all types of applicable technologies. Product critiques, which are essentially brief evaluations from experienced technology users, are collected. Quantitative evaluations, which are detailed, comparable, and objective, are performed on the most promising tools, methods, or processes.

1.2.2 Information Exchange

This objective involves exposing potential customers to available technologies and, conversely, customer requirements to technology developers. Referring to the Adoption Curve, this objective focuses on contact, awareness, and understanding. STSC products that accomplish this objective include *CrossTalk* (a monthly technology report), the annual

Software Technology Conference, specific technology reports, and electronic customer services.

1.2.2.1 CrossTalk

Over 9,000 software professionals receive *CrossTalk* monthly. This publication provides a forum for the exchange of ideas. Articles cover leading edge, state-of-the-art, and state-of-the-practice processes and technologies in software engineering.

1.2.2.2 Software Technology Conference

The annual Software Technology Conference is held each April in Salt Lake City, Utah. This conference brings together over 1,000 software professionals from government, industry, and academia to share technology solutions and exchange ideas and information.

1.2.2.3 Technology Reports

STSC Technology Reports provide detailed information on specific technologies in software engineering. This report is an example of a technology report. The current list of reports includes:

- Test Preparation, Execution, and Evaluation (TPEE 93)
- Documentation [DOC 93]
- Project Management [PM 92]
- Requirements Analysis and Design [RAD 92]
- Reengineering [RE 92]
- Source Code Static Analysis [SCSA 92]
- Software Engineering Environments [SEE 92]

These reports provide awareness and understanding of each topic in preparation for evaluation and selection of corresponding technologies. Over 7,000 of these reports have been distributed.

1.2.2.4 Electronic Customer Services

Along with the services mentioned above, the STSC also provides customers with electronic access to information via Electronic Customer Services (ECS). ECS includes a bulletin board system which is available to obtain additional information, leave messages, add information, and confer electronically. In addition, a computerized database of practice, process, and technology information is coming on-line. ECS can be accessed via the INTERNET at address 137.241.33.1 or stscbbs.af.mil or by calling 801-774-6509 with modem at 2400 or 9600 baud, 8 bit word, 1 stop bit, and no parity.

1.2.3 Technology Insertion Projects

STSC technology insertion projects are customer oriented projects that evaluate, select, and pilot the use of new processes, methods, and technologies for a specific customer. These projects can include process definition, process improvement, methodology insertion, tool insertion, and development of a technology road map. Referring to the Adoption Curve, Figure 1-1, an insertion project helps cement understanding of a process or technology, tailors an evaluation of the process or technology for the customer, and pilots the use of that process or technology with appropriate levels of training. Customers move closer to adoption of the process or technology through hands-on experience. It is important to try out technology improvements in a pilot project to confirm that the technology is appropriate for the organization and that the organization is ready and able to adopt the new technology.

1.2.4 STSC Associates

Fowler and Przybylinski [Fowler 88] propose that transitioning new technologies from a developer to a consumer requires an advocate to push the technology and a receptor to pull the technology into an organization. This concept is illustrated in Figure 1-2.

Effective change comes from within the organization. The STSC Associates objective is to develop technology receptors within individual Air Force SDSAs. These receptors, STSC Associates, are trained in the use of the STSC's information, products, and services to enhance their organization's ability to incorporate advanced practices, processes, and technologies.

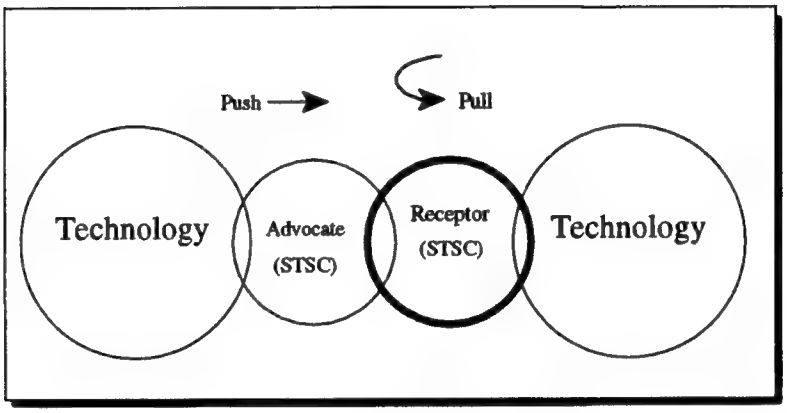


Figure 1.2 Transferring Technology

Referring to the Adoption Curve in Figure 1-1, STSC Associates complete the trek to institutionalization. Associates coming from within the organization should be politically astute and aware of internal organizational requirements. They have the highest probability of influencing the adoption and daily use of effective business practices, processes, and technologies.

2. SOFTWARE ESTIMATION

2.1 Introduction

The Project Management Tool Report, published by the STSC, describes the basic principles and capabilities of project management [PM 92]. It also discusses the importance of project planning and project pontrol (ie. scheduling, resource management, tracking, and reporting). One of the steps in project planning is determining resources needed, effort required, task duration, and overall cost for completing the proposed project.

This Software Estimation Technology Report is written to provide software project managers with a guide for estimating project cost and size, defining associated tasks, and determining a schedule sufficient to perform the project. It also contains information about the software estimation process, associated methodologies, and available software estimation tools.

Section 2.2 provides an overview of the software estimation process, introduces Software Life Cycle Models and discusses applicable government standards. Section 2.2.1 discusses the importance of clear project objectives and requirements. Section 2.2.2 discusses the importance of planning out the software estimation process, structured approaches to defining tasks, and the Work Breakdown Structure (WBS). Section 2.2.3 identifies key risks that can have a major effect on software development projects. Section 2.2.4 presents several methodologies normally used to support the software estimation process. Section 2.3 specifically identifies applicable government development standards and impacts to software estimation. Section 2.4 discusses the benefits realized from accurate software estimations.

2.2 Software Estimation Process

One of the most challenging tasks in project management is accurately estimating needed resources and required schedules for software development projects. The software estimation process includes estimating the size of the software product to be produced, determining which functions the software product must perform, estimating

effort required, developing preliminary project schedules, and finally, estimating overall cost of the project.

Size and number of functions performed are considered major productivity factors during the software development process. Effort is divided into labor categories and multiplied by labor rates to determine overall costs. Therefore, software estimation is sometimes referred to as software cost estimation.

Software life cycle models identify various phases and associated activities required to develop and maintain software, and provide excellent input into the software estimation process. Some of the more common and accepted life cycle models include: (1) Waterfall Model; (2) Rapid Prototyping; (3) Incremental Development Model; (4) Spiral Development Model; (5) Reusable Software Model; and (6) the Transform Model [Boehm 88 and Davis 88]. These models form a baseline from which to begin the software estimation process and should be reviewed and tailored to the proposed project.

Software identified as mission-critical and developed for the United States government must be developed in accordance with DOD-STD-2167A, "Defense System Software Development" [DOD 88]. This standard establishes uniform requirements for software development, and does not specify or discuss any particular method of software development. However, it requires the inclusion and documentation of the following major software development life cycle activities:

- System Requirements Analysis/Design
- Software Requirements Analysis
- Preliminary Design
- Detailed Design
- Coding and Computer Software Unit (CSU) Testing
- Computer Software Component (CSC) Integration and Testing
- Computer Software Configured Item (CSCI) Testing
- System Integration and Testing

The standard also requires that reviews and audits be held in accordance with MIL-STD-1521B, "Technical Reviews and Audits for Systems, Equipment, and Computer Programs" [DOD 85].

Additionally, structured approaches to sub-task identification are extremely beneficial in determining tasks and the required effort for each task. The WBS is a method which strongly supports this process. This method is discussed below.

The software estimation activity should be approached as a major task and therefore should be well planned, reviewed, and continually updated. The basic steps required to accomplish software estimation are described in the following paragraphs.

2.2.1 Define Project Objectives and Requirements

The objectives and requirements of a software project are usually established by upper management directive or by a contract Statement of Work (SOW). A clear set of objectives must be established in order to accurately identify project requirements. Project requirements must also include specifications that must be met and applicable standards that must be followed. Project objectives and requirements must be defined as clearly and precisely as possible in order to accomplish the project correctly, as well as identify tasks and ultimately estimate costs as accurately and early as possible.

2.2.2 Plan the Activities

As previously mentioned, the software estimation activity should be planned as a major task. The plan should detail the purpose, products, schedules, responsibilities, procedures, required resources, and assumptions made. The plan should include which estimation methodologies, techniques, and tools will be used.

The project should be organized into a hierarchical set of tasks to the lowest level of detail that available information will allow. Additionally, a breakdown of documentation requirements and associated tasks should be defined. One method available which aids this effort is the detailed WBS.

The WBS helps establish a hierarchical view and organization of the project. The top level is the software system or final software product, and subsequent levels help identify tasks and associated sub-tasks and are used to define and encapsulate system functions. Each of these tasks are divided into software development phases such as design, code and test, and integration. All activities associated with each level must be

defined including; project planning and control, configuration management, product assurance and documentation.

In addition to early development of detailed knowledge about the project, the WBS provides an excellent methodology for project data collection, tracking, and reporting. During development of the project, each of the WBS tasks can be given a project budget, and a job number which is used for reporting time spent on each project phase or activity. This provides an excellent project tracking and history data collection method. Most government contracts require that such a Cost/Schedule Control System Criteria (C/SCSC) be established. When the data are collected to an established WBS, the information can be placed in a database to be used in refining, tailoring, and customizing the software estimation process. This information becomes an invaluable input to the software estimation process for future projects.

Software project tasks/sub-tasks should be defined to the smallest component possible. All technical aspects of the project should be understood as fully as possible since the more detail known about the project the more accurate the estimates will be. Newer methodologies are evolving which aid software developers in identifying various functions needed to support the project, such as Object-Oriented Analysis and Design (OOA,OOD), therefore, organizations should actively keep abreast of evolving technologies.

2.2.3 Software Estimation Risks

The effects of inaccurate software estimation and schedule overruns are well known. The problem stems from an inability to accurately assess risks associated with various software development projects. Software estimation errors generally result from four major risk areas, which are:

(1) The inability to accurately size the software project. This results in poor implementations, emergency staffing, and cost overruns caused by underestimating project needs.

(2) The inability to accurately specify a development environment which reflects reality. This results in defining cost drivers which may be inappropriate, underestimated, or overestimated.

(3) The improper assessment of staff skills. This results in misalignment of skills to tasks and ultimately miscalculations of schedules and level of effort required, as well as either underestimating or overestimating project staffing requirements.

(4) The lack of well defined objectives, requirements, and specifications, or unconstrained requirements growth during the software development life cycle. This results in forever changing project goals, frustration, customer dissatisfaction, and ultimately, cost overruns.

All potential risks associated with the proposed software development project should be defined and weighed, and impacts to project costs should be determined. This information should always be included in the software estimation process.

2.2.4 Use Several Estimation Methodologies

Several methods may be used during the software estimation process. No one methodology is necessarily better than the other, in fact, their strengths and weaknesses may be complimentary to each other. It is recommended that more than one software estimation methodology be used for comparison and verification purposes. One method may overlook system level activities such as integration, while another method may have included this, but overlooked some key post-processing components. Five of the methods discussed in Dr. Boehm's book "Software Engineering Economics" [Boehm 81], are: analogy, bottom-up, top-down, expert judgment, and algorithms.

These methods are often used in conjunction with each other and have been used for many years by managers of software projects without the use of any formal software estimation tools. Software estimation tools have only recently been developed which incorporate these methods, and many incorporate multiple methodologies. Refer to Section 3 for a discussion of software estimation tools.

2.2.4.1 Analogy Method

Estimation by analogy means comparing the proposed project to previously completed similar projects where project development information is known. Actual data

from the completed projects are extrapolated to estimate the proposed project. Estimating by analogy can be done either at the system level or the component level.

The main strength of this method is that the estimates are based on actual project data and past experience. Differences between completed projects and the proposed project can be identified and impacts estimated. One problem with this method is in identifying those differences. This method is also limited because similar projects may not exist, or the accuracy of available historical data is suspect. Also, many projects for DOD weapon systems may not have historical precedents.

2.2.4.2 Bottom-up Method

Bottom-up estimation involves identifying and estimating each individual component separately, then combining the results to produce an estimate of the entire project.

One advantage of this method is that it provides a more detailed basis for estimation because it deals with low-level components; and therefore, tends to be more accurate. Also, it supports project tracking more directly because estimates usually address each activity within each phase of the software development life cycle. This method also promotes individual responsibility when each component is estimated by the person responsible for its development.

It is often difficult to perform a bottom-up estimate early in the life cycle process because the necessary information may not be available. This method also tends to be more time consuming and may not be feasible when either time or personnel are limited.

2.2.4.3 Top-down Method

The top-down method of estimation is based on overall characteristics of the software project. The project is partitioned into lower-level components and life cycle phases beginning at the highest level. This method is more applicable to early estimations when only global properties are known.

Advantages include consideration of system-level activities (integration, documentation, project control, configuration management, etc.) many of which may be

ignored in other estimation methods. The top-down method is usually faster, easier to implement, and requires minimal project detail. However, disadvantages are that it is less accurate and tends to overlook lower-level components and possible technical problems. It also provides very little detail for justifying decisions or estimates.

2.2.4.4 Expert Judgment Method

Expert judgment involves consulting with human experts to use their experience and understanding of the proposed project to provide the estimate.

The obvious advantage of this method is the expert can factor in differences between past project experiences and requirements of the proposed project. The expert can also factor in project impacts caused by new technologies, applications, and languages. Expert judgment always compliments other estimation methodologies. One disadvantage is that the estimates can be no better than the expertise and judgment of the expert. It is also hard to document the factors used by the expert who contributed to the estimate.

2.2.4.5 Algorithmic Method

The algorithmic method involves the use of equations to perform software estimates. The equations are based on research and historical data and use such inputs as source lines of code (SLOC), number of functions to perform, and other cost drivers such as language, design methodology, skill-levels, risk assessments, etc.

Advantages of this method include being able to generate repeatable results, easily modifying input data, easily refining and customizing formulas, and better understanding of the overall estimation methods since the formulas can be analyzed. However, the results are questionable when estimating future projects which use new technologies, and equations are generally unable to deal with exceptional conditions such as exceptional personnel, exceptional teamwork, and an exceptional match between skill-levels and tasks. Additionally, algorithms are usually developed within companies for internal use and may be proprietary as well as more reflective of the company's performance characteristics.

2.3 Standards Affecting Software Estimation

Very often the government requires software development to follow DOD-STD-2167A, which is the Department of Defense standard that specifies the overall process for the development and documentation of mission-critical software systems [DOD 88]. This standard also requires technical reviews and audits to be conducted in accordance with DOD-STD-1521B [DOD 85].

Other standards that may affect the estimation process are: MIL-STD-499A, Engineering Management, MIL-STD-490A, Specification Practices, MIL-STD-480B, Configuration Control - Engineering Changes, Deviations and Waivers, DOD-STD-1703, Software Products Standards. Software developed in accordance with these standards generally requires more resources and is more time consuming. Therefore, the software estimation process must include and adjust for these requirements where applicable.

2.4 Benefits

When the software estimation process is performed correctly, the benefits realized far outweigh the cost of doing the estimation. Some of the major benefits include lowering the cost of doing business, increasing the probability of winning new contracts, increasing and broadening the skill-level of key staff members, acquiring a deeper knowledge of the proposed project prior to beginning the software development effort, and understanding, refining, and applying the proper software life cycle model.

As these software estimation components are enhanced, refined, and continually applied, the software estimation process, associated tools, and resulting products attain higher levels of quality and ultimately benefit all.

3. STATE OF SOFTWARE ESTIMATION TOOLS

As mentioned in Section 2, one of the critical problems facing software development project managers is determining accurate software estimations for level of effort, schedules, SLOC, and overall costs. Since trends indicate that the cost of producing software products is escalating and consuming an ever increasing percentage of budgets, the need to quickly generate more reliable estimates is becoming even more important.

For many years, project managers have relied on software development teams to estimate the cost of producing software products. This has always been a subjective and intuitive process influenced by such factors as personality, opinions, and pressure to win contracts. This has encouraged low estimates and short schedules, the results of which have been devastating to companies and projects, including projects of national importance.

For these reasons, software estimation tools have been developed since the late 1970s to provide a better defined and more consistent software estimation process. These tools have been developed from historical data collected from thousands of software projects, as well as research performed to identify key productivity factors. Early tools were hampered by the scarcity of reliable data, however, as more data became available, estimation tools were improved and continue to evolve. Most software estimation tools use algorithms, and some of the more advanced tools are rule-based or knowledge-based as well as interactive.

Good software estimation tools do not always guarantee reliable software estimates. If inaccurate software size estimations and attribute ratings are input, then inaccurate estimates will result. Additionally, organizations need to customize the software estimation tools to their own development environment.

This requires collecting and maintaining historical data from current and past projects to provide the necessary inputs required for the software estimation tools. The software development organization should establish a staff that is thoroughly trained in the software estimation process and available estimation tools, and they should do all software

estimation activities. Experience and existing tools dictate what project development information needs to be maintained.

This section provides an overview of available software estimation tools. Section 3.1 discusses the major functional capabilities which software estimation tools should perform, and Section 3.2 discusses input data required to support the use of those tools. Section 3.3 provides an analysis of software estimation tools. Section 3.3.1 presents tools based on the COCOMO model. Section 3.3.2 presents other Cost Estimation tools. Section 3.3.3 presents Size Estimation tools. The remaining sections describe the information contained in the appendices.

3.1 Desired Functional Capabilities

Major functional capabilities that should be considered when selecting a software estimation tool are listed below. Depending on the organization's needs, the level of significance of these capabilities may differ, and should be considered accordingly. In addition, the organization should analyze their own needs and identify additional desired capabilities specific to them. The organization should then match available tools with overall needs.

In general, the tool should:

(1) Allow easy adaptation to an organization's development environment - This means the tool needs to be capable of being customized to fit the using organization's development environment. Customization includes allowing the developer to define applicable inputs, as well as modify coefficients and exponents of the equations used by the tool. This feature will allow continuous improvement to the estimation capability of the tool since the organization's historical data and current project data will be included in the software estimate generated.

(2) Be relatively easy to learn and use - The tool should be well documented including methodologies and equations used. Documentation should be at a level that is understandable. The tool should include help menus and examples sufficient to assist the support staff in answering questions and

providing training. The amount of formal training required to use the tool should be relatively minimal, required inputs should be well defined, and visibility of internal equations and theories should be provided.

(3) Provide early estimates - The tool should be capable of generating estimates early in the life cycle process when requirements and development environments are not fully defined. The tool should also allow task detail to be added incrementally as functions, activities, and other information becomes more defined. Since there are many unknowns early in the estimation process, the tool should reflect degrees of uncertainty based on the level of detail input. In general, the tool should provide sufficient information to allow initial project resource planning as well as reasonably early "go/no go" decisions.

(4) Be based on software life cycle phases and activities - The tool should be capable of providing estimates for all phases and activities of the most commonly used software life cycle models. It should also allow the organization to decompose and map software development tasks into those phases and activities, as well as support a WBS. In addition, it should allow for "what if" situations and include factors for design trade-off studies.

(5) Allow for variations in application languages and application function - It is very important that the tool provide estimates specific to the application of the software project since the associated estimation equations, cost drivers, and life cycle phases should be unique to each application area. General application categories include Information Systems (IS), Simulation and Modeling Systems, Real-Time Systems, Accounting Systems, and systems based on higher-order languages.

(6) Provide relatively accurate size estimates - The size of a software development project is a major cost driver in most estimation tools, yet size is one of the most difficult input parameters to estimate accurately. The tool should include the capability to help estimate the size of the software development project, or at least help define a method for estimating the size.

(7) Provide relatively accurate schedule estimates - As previously mentioned, schedule overruns are common and can be extremely costly. The software estimation tool should be able to provide schedule estimates accurately. The purpose of scheduling is not only to predict task completion given task sequence and available resources, but also to establish starting and ending dates for the associated work packages and life cycle phases.

(8) Provide maintenance estimations separately - The software estimation tool should be able to provide software maintenance estimations as a separate item. Software maintenance includes such things as correcting errors, modifying the software to accommodate changes in requirements, or extend and enhance software performance.

3.2 Input Data Collection

A very important aspect of software estimation is data collection. Data must be collected for both inputs to the tool and tool verification, validation, and customization.

Estimates generated by the tools are only as good as the input data used. Careful analysis of all tool inputs are essential since small changes in input values can result in large variations in overall cost and schedules.

Inputs vary between tools. Before using a tool, review input requirements and information collected, documentation, and examples provided with the tool. When possible, discuss these with individuals familiar with the tool.

Using historical data as a basis for customizing a tool is essential. Obviously, this can only be accomplished when historical data is available. Insure that information for current project development efforts are saved for future reference.

At a very minimum, use software life cycle model phases and activities as a basis for collecting and maintaining project development information for future tool use.

3.3 Analysis of Software Estimation Tools

There are many software estimation tools on the market today that provide software estimation support. Some tools estimate the size of the software project, while others use size as an input and provide estimates of effort, schedule and cost.

None of the tools presented in this report are necessarily recommended over other tools since each one has unique capabilities and limitations. Therefore, organizations considering using estimation tools should review available tools, analyze needs, and then determine which tools are most appropriate to their application and development environment.

Most software estimation tools use algorithms, and most are proprietary, however, the COnstruction COst MOdel (COCOMO) is widely used as a baseline for many estimation tools because it is considered an "open model" since all details are well documented and have been published. For this reason, and because there are so many computerized estimation tools available that use COCOMO, a separate section on these tools is provided, and a detailed overview of COCOMO is also provided in Appendix F.

3.3.1 COCOMO-based Estimation Tools

The following are COCOMO based estimation tools. For more product information about these tools see Appendix B.

3.3.1.1 CB COCOMO

Crystal Ball (CB) COCOMO is distributed by Decisioneering Incorporated. This tool estimates time and cost of software development projects, and allows entering of actual project data at various phases in the software life cycle. This tool requires Decisioneering's forecasting and risk analysis tool, Crystal Ball, and operates on Macintosh systems.

3.3.1.2 COCOMOID

COCOMOID is distributed by the Air Force Cost Center (via electronic bulletin board) and the Society for Cost and Economic Analysis (SCEA). COCOMOID is a complete COCOMO implementation supporting all known published COCOMO specifications. COCOMO models used include Basic, Intermediate, Detailed, and Maintenance models. It also includes Enhanced Ada, Ada Process, and Incremental Development models. This tool operates on PC compatible systems.

3.3.1.3 COCOMO1

COCOMO1 was developed by Level Five Research, Inc. and is marketed by Solar Powered Emergency Communications Systems (SPECS), Inc. It is an expert system-based software tool that estimates cost and time requirements for software projects. Through a series of questions COCOMO1 assists in determining the proper COCOMO cost model, mode, and effort coefficients. This tool uses fifteen development and maintenance cost drivers and applies formulas from the COCOMO model to these factors. COCOMO1 runs on all PC compatible systems.

3.3.1.4 CoCoPro

CoCoPro is distributed by Iconix Software Engineering, Inc. It estimates resources needed to complete software development projects using the COCOMO model. This tool uses exponential functions and attributes to calculate development costs. Inputs allowed include personnel experience and capabilities, project complexity, product factors, and hardware limitations. CoCoPro operates on a Macintosh.

3.3.1.5 COSTAR

COSTAR is an interactive software cost estimation tool marketed by Softstar Systems. It is a full implementation of the detailed COCOMO model and offers side-by-side comparisons of several alternative estimates. This tool also provides automatic recalculation and display of results, and uses definable cost drivers. COSTAR version 3.0 includes support of Ada COCOMO. This tool also uses Function Point Analysis (FPA) for software size estimations. It is available for both PC compatible and VAX computer systems. See Appendix G for a discussion of FPA.

3.3.1.6 COSTMODL

COSTMODL was developed by the Software Technology Branch, Spacecraft Software Division, NASA/Johnson Space Center [NASA/JSC 91], and provides estimates for effort, cost, and schedule. It implements all the COCOMO models except the detailed model. In addition, it includes a simplified linear model using productivity data from completed NASA projects. This model is also called the "Keep It Simple, Stupid" (KISS) model.

COSTMODL is presently used at over 100 government, military, and contractor sites, as well as NATO headquarters, the British Ministry of Defense, and several universities in the United States and England. It has been selected as the standard cost estimating tool for NASA's Space Station Freedom Software Support Environment.

COSTMODL contains five different models for estimating non-Ada and Ada products, and products which are to be delivered as a series of incremental development phases. All of the parameters defining each of the models are accessible to the user. Basic estimating equations can be calibrated to the user's software development environment and type of product. Also, the set of factors which influence software development costs can be redefined.

Given the data describing the software development productivity experience for a user's organization, COSTMODL will automatically compute the coefficients and exponents which will provide the most meaningful estimates for new products to be developed within that organization. It also contains an extensive set of linked, context-sensitive help displays and demonstration files designed to quickly familiarize the new user with its operation.

This tool is free to both employees and contractors of NASA, as well as other government agencies, and has been submitted to NASA's Computer Software Management and Information Center (COSMIC) for distribution into the private sector. COSTMODL currently runs on PC compatible systems.

3.3.1.7 GECOMO Plus

GECOMO Plus is marketed by GEC-Marconi Software Systems. It is a special enhancement of the COCOMO model and uses 17 cost drivers. It provides cost estimations for both non-Ada and Ada projects.

GEC-Marconi markets a companion tool for size estimation called SIZE Plus (see Section 3.3.3.7). Both GECOMO Plus and SIZE Plus are X-Windows/OSF Motif compatible and are available for both the UNIX and VMS operating systems.

3.3.1.8 GHL COCOMO

GHL COCOMO is marketed by GHL Associates, Inc. and features three levels of detail: multiproject, data retention, and sensitivity analysis. This tool also allows for "what-if" scenarios. It operates on PC compatible systems.

3.3.1.9 REVIC

The Revised Enhanced Version of Intermediate COCOMO (REVIC) was developed by Mr. Raymond L. Kile of Hughes Aerospace. The Air Force Contract Management Division, Air Force System Command, Kirtland Air Force Base, New Mexico, sponsored the development for use by its contract administrator [Kile 91].

The main difference between REVIC and COCOMO is the coefficients used in the effort equations. REVIC changed the coefficients based on using a database of recently completed DOD projects. It also uses a different method of distributing effort and schedule to each phase of product development, and applies an automatic calculation of standard deviation for risk assessment.

REVIC provides a single-weighted "average" distribution for effort and schedule along with the ability to let the user vary the percentages in the system engineering and development test and evaluation phases. REVIC employs a different Ada model than Ada COCOMO. The REVIC model has also been enhanced by using a Program Evaluation and Review Technique (PERT) statistical method for determining the lines of code to be developed. See Section 3.3.3 for a brief discussion of PERT.

In addition to providing estimates for cost, manpower, and schedule, the program creates estimates for typical DOD-STD-2167A documentation sizing and long term software maintenance. REVIC's schedule estimates are often considered lengthy because it assumes that a project's documentation and reviews comply with the full requirements of DOD-STD-2167A. REVIC operates on PC compatible systems.

3.3.1.10 SECOMO

Software Engineering Cost Model (SECOMO) is available at no cost from IIT Research Institute in Rome, NY. Enhancements include an improved user interface, on-line help, and an expanded user's manual. Version 7.0 will operate on a PC compatible system or a VAX/VMS 5.2 or later operation system.

3.3.1.11 SWAN

The Software Analysis (SWAN) cost model was developed by IIT Research Institute for the U.S. Army Program Manager for Training Devices (PMTRADE) in the Ada programming language. SWAN is available at no cost to government agencies and associated contractors.

This tool supports the intermediate version of COCOMO, including Ada COCOMO with full three-level software hierarchy support. SWAN utilizes FPA techniques to determine software size estimates. It provides estimates for software development from requirements analysis through integration and test, as well as estimates for up to 5 years of maintenance. SWAN runs on PC compatible systems under MS-DOS 3.1 or later.

3.3.2 Other Software Estimation Tools

This section contains a discussion of additional software estimation tools. For more information about these tools see Appendix B.

3.3.2.1 CA-ESTIMACS

CA-ESTIMACS is part of a family of tools called CA-UNIPACK/PEP marketed by Computer Associates International, Inc. CA-UNIPACK/PEP consists of four tools: CA-ESTIMACS, CA-PLANMACS, CA-ADVISOR, and CA-SuperProject.

CA-ESTIMACS uses research drawn from a data base of more than 14,000 completed software projects. It develops estimates at or before the requirements definition phase of the software life cycle. This tool allows for early "what-if" analyses of alternative life cycle strategies.

A companion tool named CA-FPXpert is also distributed by Computer Associates and is used to estimate the size of the software product (see Section 3.3.3.2). All of these tools operate on PC compatible systems.

3.3.2.2 CHECKPOINT and SPQR/20

CHECKPOINT and SPQR/20 (Software Product, Quality, and Reliability Twenty-Questions) are software estimation tools distributed by Software Productivity Research (SPR).

CHECKPOINT is a knowledge-based software estimation tool that has largely superseded SPQR/20. It's algorithms are derived from measurements of more than 4200 software projects, and it is applicable to all phases of the software development life cycle. It applies to all types of programs and incorporates Function Points or Feature Points to calculate the size of a software product. Feature points are SPR's method of measuring functionality.

SPQR/20 is based on the work of Capers Jones [Jones 86] and incorporates proprietary algorithms. It was one of the first models to use function points as a measure of size to estimate source lines of code. Most of the inputs define experience level, development method, and development environment. Other inputs include project type and class. SPQR/20 estimates maintenance support for up to a five-year period [SPR 86]. CHECKPOINT and SPQR/20 operate on PC compatible systems.

3.3.2.3 COSTEXPERT

COSTEXPERT is a software estimation tool that uses expert system technology. It was developed by the Institute for System Analysis, Inc. (ISA) and is marketed by Technology Applications/Engineering Corporation. It asks questions about the functionality of the software being developed [ISA 90].

COSTEXPERT directly estimates software-related efforts such as program management, security, and configuration management. It supports multiple languages and different development standards. It also takes into account software reuse.

A companion tool named SIZEEXPERT is also distributed by Technology Applications/Engineering Corporation and is used to estimate the size of the software product based on the COSTEXPERT questions. (see Section 3.3.3.4). These tools operate on PC compatible systems.

3.3.2.4 MicroMan ESTI-MATE

MicroMan ESTI-MATE is an estimating and planning tool for Information Systems oriented projects. It uses FPA methodologies, and is distributed by POC-IT Management Services, Inc.

MicroMan ESTI-MATE provides a breakdown of the hours required for all phases, activities, and tasks that make up a project. It is fully integrated with the MicroMan II Project and Staff Management System tool used for scheduling, tracking, and reporting. For information about MicroMan II, refer to the STSC Project Management Tools Report [PM 92]. MicroMan ESTI-MATE operates on PC compatible systems.

3.3.2.5 PRICE S

The PRICE S tool is distributed by GE PRICE Systems. This tool was first developed in 1977, and is considered one of the first complex commercially available tools used for software estimation [Freiman-Park 79]. Equations used by this tool are

proprietary. However, descriptions of the methodology scheduling algorithms used can be found in a paper published by GE PRICE Systems [Price 88].

The PRICE S tool is based on Cost Estimation Relationships (CERs) that make use of product characteristics in order to generate estimates. CERs were determined by statistically analyzing completed projects where product characteristics and project information were known. The major input to PRICE S is SLOC. Software size may be input directly or automatically calculated from quantitative descriptions. PRICE S also permits function points to be input as an alternative to SLOC. Other inputs include software function, operating environment, software reuse, complexity factors, productivity factors, and risk analysis factors. Successful use of the PRICE S tool depends on the ability of the user to define inputs correctly. It can be customized to the needs of the user. It is now available for Windows and Unix/Motif.

3.3.2.6 PROJECT BRIDGE

PROJECT BRIDGE Planning and Estimating System is marketed by Applied Business Technology Corporation. It is a knowledge-based tool used for profiling, estimating, and planning projects in a software engineering environment. It allows users to produce estimates based on Function Points or an organization's own estimating factors. This tool integrates with the Project Workbench project management tool for Information Systems projects. For more information about Project Workbench refer to the STSC Project Management Tools Report [PM 92]. PROJECT BRIDGE operates on PC compatible systems.

3.3.2.7 SASET

The Software Architecture, Sizing and Estimating Tool (SASET) was developed for DOD by the Martin Marietta Corporation. SASET is a forward-chaining rule-based expert system utilizing a hierarchically structured knowledge data base. The data base is composed of projects with a wide range of applications [Silver 90].

SASET provides functional software sizing values, development schedules, and associated manloading outputs. It provides estimates for all types of programs and all phases of the development cycle. It also provides estimates for maintenance support and performs a risk assessment on sizing, scheduling, and budget data.

SASET uses a five-tiered approach for estimation including class of software, source lines of code, software complexity, maintenance staff loading, and risk assessment. The user can either input the program size directly or allow SASET to compute size, based on function-related inputs. The tool also has an extensive customization file in which the user can adjust many parameters. It operates on PC compatible systems.

3.3.2.8 SEER-SEM

System Evaluation and Estimation of Resources - Software Estimation Model (SEER-SEM) is distributed by Galorath Associates and is currently under a five year Air Force wide license agreement. It provides software estimations with knowledge bases developed from many years of completed projects [Galorath 92].

The knowledge base allows estimates with only minimal high level inputs. The user need only select the platform (i.e. ground, unmanned space), application (i.e. command and control, diagnostic), development methods (i.e. prototype, incremental), and development standards (i.e. 2167A). SEER-SEM is applicable to all types of software projects and considers all phases of software development.

A companion tool called SEER-Software Size Model (SSM) is also distributed by Galorath Associates and is used to estimate the size of the software product (see Section 3.3.3.5).

SEER-SEM is designed to run on PC compatible systems running Microsoft Windows 3.0/3.1 (Air Force license includes MS-DOS version). It is also available for the Apple Macintosh running system 6.0.3 and above and the UNIX/SUN workstation.

3.3.2.9 SLIM

The Software Life Cycle Model (SLIM) is marketed by Quantitative Software Management (QSM). SLIM was developed in 1979 by Mr. Larry Putnam [Putnam-Fitzsimmons 79]. Originally developed from analyses of ground-based radar programs, the SLIM tool has been expanded to include other types of programs. It can be customized for the user's development environment [QSM-SLIM 87].

SLIM supports all phases of software development, except requirements analysis, as well as all sizes of software projects, but was especially designed to support large projects.

Success in using SLIM depends on the user's ability to customize the tool to fit the software development environment, and to estimate both a Productivity Index (a measure of the software developer's efficiency) and a Manpower Buildup Index (a measure of the software developer's staffing capability). SLIM also provides a life-cycle option which extrapolates development costs into the maintenance phase.

A companion tool named SIZE PLANNER is also distributed by QSM and is used to estimate the size of the software product (see Section 3.3.3.6).

QSM provides a training course and leases the tool via a time sharing service. There is also a PC compatible version of SLIM that can be leased for a yearly fee.

3.3.2.10 SOFTCOST-R

SOFTCOST-R is a software estimating tool developed by Reifer Consultants, Inc. (RCI) [RCI 89]. SOFTCOST-R is based upon the pioneering modeling work done by Dr. Robert Tausworthe of the Jet Propulsion Laboratory [Tausworthe 81]. It contains a data base of over 1500 data processing, scientific and real-time programs. A key input is SLOC, which can be input directly or computed from Function Points. SOFTCOST-R is applicable to all types of programs, however, it was specifically configured to estimate real-time and scientific software systems, and considers all phases of the software development cycle.

The tool's primary input is SLOC, however, it also uses the same inputs and provides the same outputs as COCOMO which allows direct comparisons to be made.

SOFTCOST-R has some unique inputs such as use of peer reviews, customer experience, and degree of standardization. It also supports a standard WBS for task planning and scheduling.

RCI provides SOFTCOST-Ada, which is a tool to estimate Ada and C++ development costs. SOFTCOST-Ada is a cost estimation tool specifically developed to estimate systems using object-oriented techniques.

RCI also has a separate size estimation tool called ASSET-R to estimate the size of the software product (see Section 3.3.3.1).

SOFTCOST-R, SOFTCOST-Ada, and ASSET-R are leased on an annual license basis, and require a PC compatible running DOS 2.3 or higher.

3.3.2.11 SYSTEM-4

SYSTEM-4 is marketed by Computer Economics, Inc. (CEI). It contains a proprietary model that is based on the work of Jensen, Boehm, Putnam, and other noted software experts [Jensen 81].

SYSTEM-4 is applicable to all types of programs and all phases of the software life cycle. Inputs consist of size (SLOC), twenty environmental factors, seven development factors, software type, and constraints. This tool comes with 23 pre-defined default parameter files. The default sets provide typical values for all parameters except size. There are also seven parameter subset files for various implementations of DOD-STD-1703, DOD-STD-2167A, and varying degrees of Ada experience.

The user only needs to select one of the default sets and input the SLOC estimate to perform a quick estimate. SYSTEM-4 can accommodate multiple CSCIs or tasks, and each task can be broken down into elements and units. There is a limit of 64 tasks, 64 elements, and 64 units. SYSTEM-4 can be customized to reflect the user's software development environment [CEI 89].

CEI has a companion software size estimating tool called Computer Economics Incorporated Sizing (CEIS) System (see Section 3.3.3.3). These tools operate on PC compatible systems.

3.3.3 Software Size Estimation Tools

As discussed previously, a very important factor in estimating software development projects is the ability to estimate the size of the product. Many software estimating tools use size in SLOC or functions performed as the major input. Size is also considered by software development project managers as a major technical performance or productivity indicator that allows them to track the project during software development.

The most commonly used method to estimate the size of a software product is by using both expert judgment and the analogy method. The experts determine the functions the software will perform and estimate the size by comparing the new system to completed projects similar in characteristics.

Software size estimation tools use basically three estimation methods. They are the Analogy Method, Expert Judgment Method, and Algorithmic Method. These methods are very often used together.

Estimation tools using analogy compare the new program to similar programs of known size. Because past projects are not always exactly like the new project, the estimate is adjusted by a factor determined from experience. These tools accept characteristics of new programs as input, then search a data base for similar programs. The tools either list the similar programs or provide an estimate of size based on an average of the size of the similar programs selected from the data base.

Expert judgment tools use the opinion of one or more experts to estimate the size of the program, or use structured questions designed to extract judgment from the experts. These are the rule-based or expert system tools.

Many tools use the algorithmic method by applying equations to determine size estimations. A technique that is becoming very widely used is FPA. Because so many software estimation tools use this method, it is presented in more detail in Appendix G.

One problem with FPA is that it was developed for IS-oriented programs and does not take into consideration the number or complexity of algorithms in scientific and real-time programs. The CHECKPOINT software estimation tool has developed a technique

to account for estimation complexities introduced by scientific and real-time programs. The technique is called Feature Points. Also, the ASSET-4 and SIZE Plus estimating tools address this problem.

Many software estimation tools such as REVIC and SLIM use extensions of the Program Evaluation and Review Technique (PERT). PERT is based on a beta distribution of estimates provided by the user and calculates expected size according to the equation:

$$\text{Expected Size} = (S + 4(M) + L)/6$$

where S, M, and L are estimates of the smallest size, most likely size, and the largest size, respectfully.

3.3.3.1 ASSET-R

ASSET-R is a function point sizing tool developed to estimate the size of data processing, real-time, and scientific software systems, and is marketed by Reifer Consultants, Inc. It utilizes a knowledge-based system which extends the theory of function points into scientific and real-time systems by considering issues like concurrence, synchronization, and reuse in its mathematical formulation. The formulas use as many as nine parameters to develop function point counts. It also couples function point and operand/operator counts with architectural, language expansion, and technology factors to generate the size estimate. ASSET-R works with RCI's SOFTCOST-R and SOFTCOST-Ada software estimation tools (see Section 3.3.2.10). It operates on PC compatible systems.

3.3.3.2 CA-FPXpert

CA-FPXpert is distributed by Computer Associates International, Inc. It uses FPA for size estimation of IS type software projects, and conforms to accepted IFPUG standard counting practices. It includes an on-line tutor to help the function point counting process. CA-FPXpert works in conjunction with CA-ESTIMACS to provide software size estimation input (see Section 3.3.2.1), and operates on PC compatible systems.

3.3.3.3 CEIS

CEIS is marketed by Computer Economics, Inc. Estimations are generated by comparing the attributes of the new project to the attributes of three reference projects of known size. The user determines any six attributes that contribute to the number of lines of code and ranks them in order of importance, then selects three reference projects of known size. Separate algorithms are used to produce four independent estimates and to determine a level of confidence. CEIS works in conjunction with SYSTEM-4 (see Section 3.3.2.11), and operates on PC compatible systems.

3.3.3.4 SIZEEXPERT

SIZEEXPERT was developed by the Institute for Systems Analysis and is marketed by Technology Application/Engineering Corporation. This tool is an expert judgment tool that produces estimates of lines of code based on questions asked by COSTEXPERT. Both tools are packaged and distributed together (see Section 3.3.2.3), and operate on PC compatible systems.

3.3.3.5 SEER-SSM

SEER-SSM is marketed by Galorath Associates and is available to government personnel and contractors under an Air Force-wide contract. It produces software size estimates in lines of code or function points. It also provides its own historical data base to save time in producing the size estimate. SEER-SSM works with SEER-SEM software estimation tool (see Section 3.3.2.8), and operates on PC compatible systems.

3.3.3.6 SIZE PLANNER

SIZE PLANNER is distributed by Quantitative Software Management, Inc. It uses four independent approaches for size estimation including Fuzzy Logic, Function Points, Standard Component, and New/Reuse/Modified sizing. Each approach views the product from a unique perspective. This capability provides a cross check for the overall estimate which reduces the uncertainty of the estimate. SIZE PLANNER is used in conjunction with SLIM (see Section 3.3.2.9), and operates on PC compatible systems.

3.3.3.7 SIZE Plus

SIZE Plus is marketed by GEC-Marconi Software Systems. This tool uses FPA to estimate the size of the software product. It supports both data processing and real-time applications. SIZE Plus provides five different methods to perform FPA. Three of these are oriented towards IS applications and the other two are used for real-time or embedded software applications. This tool is used to provide size estimates for GECOMO Plus (see Section 3.3.1.7), and is available for UNIX or VMS operating systems running X-Windows/OSF Motif.

3.4 Software Estimation Tools List

Appendix A contains a list of tools identified by the STSC at the time this report was published. Vendors are invited to contact the STSC regarding any tools that were inadvertently not included in this list.

3.5 Software Estimation Product Sheets

Appendix B contains product information provided by vendors for their particular software estimation tool, and is arranged alphabetically by tool name. More in-depth information is presented to help customers make preliminary assessments about the tool. Included are pricing, platform/operating system, description/purpose, and additional information about the tool and vendor. Vendors are invited to provide updated information or information for tools not included. Please contact the STSC for any unpublished tool product sheets that may be available.

3.6 Software Estimation Product Critiques

Appendix C contains critiques for selected software estimation tools, and is also arranged alphabetically by tool name. The authors are experienced users of the software and, in most cases, have been referred by the vendor. Product critiques highlight the use, notable strengths and weaknesses of the tool and provide advice for potential users of the tool. The critiques are the opinions of the individual users, and do not reflect the opinion of the STSC. Contact the STSC for any unpublished product critiques or updated product critiques that may be available.

3.7 References and Recommended Readings

Appendix D cites references used in this report, as well as publications that provide further information about software estimation tools.

The book "Software Engineering Economics" by Dr. Boehm [Boehm 81] is highly recommended not only for his description of the COCOMO model, but also his in-depth work in software estimation principles and practices. Another excellent write-up on software estimation is "An Introduction to Software Cost Estimation and Measurement" by Daniel V. Ferens [Ferens 1/91].

There have been several evaluation studies conducted in the past to validate selected tools. The results of these studies show that specific tools appear to be accurate for a particular environment, and certain tools appear to be accurate estimators for selected applications. The studies were limited in several aspects, and no one tool was shown to be accurate for a wide range of applications. The studies reemphasized the importance of tool customization to the software developer's environment.

Some problems with the studies are that they do not all include the same tools, they do not include the most updated versions of the tools, and they do not include some of the newer tools on the market. The following is a list of those studies:

"A Descriptive Evaluation of Automated Software Cost Estimation Models," by Elizabeth K. Bailey, [Bailey 86].

"Evaluation of Eight Software Support Cost Models," by Daniel V. Ferens, [Ferens 91].

"A Comparison of Software Schedule Estimators," by Bryan A. Daly, [Daly 90].

"An Analysis of Schedule Determination in Software Program Development and Software Development Estimation Models," by Crystal D. Blalock, [Blalock 88].

"Current Research on Schedulers for Aerospace Industry Software," by Richard M. Greathouse, and Kelly L. Shipley, [Greathouse-Shipley 90].

"Estimating the Cost of Ada Software Development," by Illinois Institute of Technology Research Institute, [IITRI 89].

3.8 Glossary

Appendix E contains acronyms used in this report and definitions of terms commonly used in software estimation technology.

This page intentionally left blank

4. TRENDS AND CONCLUSIONS

4.1 Trends

Advances in languages, development methodologies, and other areas will have to be addressed by future software cost estimating models and associated methodologies. As software development technology matures, changes in development and support concepts occur which will impact software cost estimation. Concepts such as prototyping and spiral development present a challenge to cost estimation since normal software development cycles are altered.

Artificial Intelligence (AI) represents a growing area of modern technology. Since AI is software-intensive, proper management of AI software, including cost estimation, will be a challenge for software managers. The development of software for expert system and other AI applications will probably require a different development process.

The trend for the future will include better and more accurate ways of developing software estimation methodologies for:

- Software size estimates.
- Resource estimates for maintenance or support.
- Incorporating the effects of Ada and new paradigms such as rapid prototyping and fourth-generation languages.
- Modeling the dynamic interaction of variables that affect productivity, cost, and quality.
- Object-Oriented Design.

The trend in software estimation tools is to provide a whole family of models which not only estimate cost and effort of software development, but hardware as well. The tools are being upgraded to support higher-order languages such as Ada and C++. The most significant improvement is the use of data collected from past software projects to customize the tool to the organization's environment. This is especially true within agencies of DOD and NASA.

4.2 Conclusions

This report has discussed the software estimation process and the various methodologies used in software estimation. The basic software estimation functional capabilities were also discussed. Many different software estimation tools were examined, and product sheets were provided. Various users of the tools were contacted to verify user satisfaction and to obtain their evaluations and recommendations.

A review of product literature and user manuals indicates that many tools will perform most of the functional capabilities outlined in this report. The users generally agree that the tools they are using satisfy their requirements.

The software estimation organization must be able to customize the software estimation tool to their own software development environment. This requires collecting historical data from past completed projects to accurately provide the inputs that the software estimation tool requires. The software development organization should establish a staff that is thoroughly trained in the use of the tools. This staff should do all the software estimation activities and determine what data should be collected to provide a historical data base for future reference.

The use of two or more software estimating tools using different methodologies is highly recommended. The software development organization should select a primary tool for software estimation and an alternate tool for comparison and validation. These tools should be used throughout the software development process.

Algorithmic tools are considered to be the best for software estimation for the following reasons:

- Equations are based on previous development projects.
- Outputs are repeatable and formulas can be analyzed.
- They can be customized to fit the user's environment.
- They require minimal time and effort to use.
- They are particularly useful in earlier phases of software development.
- They are most frequently used by DOD agencies.

The algorithmic tool of choice should be one that is non-proprietary because visibility into the inner workings of the tool is invaluable. The software development organization must be able to easily use and understand the tool as well as customize the tool to their development environment. The estimator can also better understand and provide the necessary inputs required by the tool.

COCOMO is well documented and considered to be the model that is easiest to use and understand. There are many software estimation tools that use COCOMO. Of these, REVIC is the most widely used in the DOD and was revised based on Air Force historical data. It also costs nothing to acquire, and there is a very active user's group that continues to verify and enhance its capabilities. COSTMODL is also highly recommended especially for users in NASA, because it was developed based on NASA historical data. It also incorporates Ada COCOMO [Boehm 87] and provides for incremental software development. COCOMOID, SECOMO and SWAN are tools based on COCOMO that are also available to the government and government contractors.

Another government tool that is highly recommended by its users is SASET. SASET uses a rule-based expert system combining expert judgment, bottom-up and algorithmic methodologies. This tool is also widely used in DOD and costs nothing for DOD personnel. SASET is considered to be more difficult to use and more difficult to customize to the user's environment compared to REVIC. However, with proper training it has been shown to be a very accurate schedule estimator [Greathouse 90].

Software size is the major input to most of the algorithmic software estimation tools. However, software size estimation continues to be a problem because no tool or method has proven to be accurate for all types of programs. FPA has been demonstrated and validated as being accurate for IS programs, but further validations need to be performed for scientific, real-time, and higher order language programs. ASSET-R, a softcost size estimation tool, contains language adjustment factors and inputs to adapt function points for scientific, real-time, and various programming languages such as Ada. CHECKPOINT and SIZE Plus also use an enhancement to FPA to estimate the size of scientific and real-time type software.

A review of previous studies (Section 3.7) shows that specific tools appear to be accurate for a particular environment and certain tools seem to be accurate for selected

applications. The studies were limited in several aspects and no one tool was shown to be accurate for a wide range of applications.

The STSC can assist customers in the selection of a software estimation process, methodology, and tools applicable to their specific software development needs. The STSC can analyze the customer's software development environment and help customize the software estimation tool to that environment.

Appendix A:

Software Estimation Products Long List

Software Estimation Products List - By Product

Product	Platform	Vendor	Classification/Comments
ASSET-R	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software size estimation - One in a family of models for estimation of software development. Uses Function Point Analysis.
CA-ESTIMACS	MS-DOS	Computer Associates Int., Inc. Calverton, MD 20705 301-937-1133	Software cost estimation - Estimates effort, schedule and cost of Information Systems software.
CA-FPXpert	MS-DOS	Computer Associates Int., Inc. Calverton, MD 20705 301-937-1133	Software size estimation - Uses Function Point Analysis methodology for size estimation.
CB COCOMO	Mac/Windows	Decisioneering, Inc. Boulder, CO 80301 303-292-2291	Software cost estimation - Based on COCOMO to estimate effort and cost of software development projects.
CEIS	MS-DOS	Computer Economics, Inc. Marina Del Rey, CA 90292 310-827-7300	Software size estimation - Calculates four independent size estimates based on comparison to other known software projects.
CHECKPOINT	MS-DOS	Software Products Research , Inc. Burlington, MA 01803 617-273-0140	Software cost estimation - Knowledge-based software estimation tool. Uses function points and feature points to estimate size.
COCOMO1	MS-DOS	Specs Inc. Junction City, OR 97448 503-998-8729	Software cost estimation - Artificial Intelligence Front End with COCOMO model.
COCOMOID	MS-DOS	Air Force Cost Center Wright-Patterson AFB, OH 45433 513-257-3927	Software cost estimation - Based on COCOMO. Provides estimates also based on enhanced, Ada, Ada process and incremental development models.
CoCoPro	Macintosh	Iconix Software Engineering, Inc. Santa Monica, CA 90405 310-458-0092	Software cost estimation - Estimates resources needed using standard COCOMO.
COSTAR	MS-DOS/VMS	Softstar Systems Amherst, NH 03031 603-672-0987	Software cost estimation - Uses detailed COCOMO, Ada COCOMO and sizing with function points.
COSTEXPERT	MS-DOS	Technology Appl./Engr. Corp. Bethesda, MD 20817 301-571-8510	Software cost estimation - "Expert System" based model. Does not use lines of code estimate or function points.
COSTMODL	MS-DOS	NASA/JSC's Software Tech. Branch Houston, TX 77058 713-483-9092	Software cost estimation - Detailed COCOMO model. Available to government by NASA/JSC.
GECOMO Plus	X-Windows OSF,Motif, Unix, VMS	GEC-Marconi Software Systems Reston, VA 22090 703-648-1551	Software cost estimation - Implements an extended detailed COCOMO model and includes Ada.
GHL COCOMO	MS-DOS	GHL Associates, Inc. Haverford, PA 19041 215-896-7307	Software cost estimation - Estimates software development costs based on COCOMO model.
Micro Man ESTI-MATE	MS-DOS	POC-IT Management Services, Inc. Santa Monica, CA 90401 310-393-4552	Software cost estimation - Estimates effort for Information Systems projects using Function Point Analysis.
PRICE S	MS-DOS	GE Price Systems Moorestown, NJ 08057 800-437-7423	Software cost estimation - One in a family of cost models for software and hardware costs, effort, and schedule. Uses functionality and source lines of code.
PROJECT BRIDGE	MS-DOS	Applied Business Technology Corp. New York, NY 10013-3992 800-444-0724	Software estimation - Estimates based on Function Point Analysis.
REVIC	MS-DOS	Air Force Cost Analysis Agency Arlington, VA 22202 703-746-5865	Software cost estimation - Uses Intermediate COCOMO with added features for life cycle costing and risk analysis. (Public domain software)

Software Estimation Products List - By Product

Product	Platform	Vendor	Classification/Comments
SASET	MS-DOS	Air Force Cost Analysis Agency Arlington, VA 22202 703-746-5865	Software cost estimation - A forward chaining, rule-based expert system utilizing a hierarchically structured knowledge database.
SECOMO	MS-DOS, VMS	IIT Research Institute Rome, NY 13440 315-336-2359	Software cost estimation - Uses full COCOMO maintenance cost estimation.
SEER-SEM	MS-DOS	Galorath Associates, Inc. Los Angeles, CA 90009 310-670-3404	Software cost estimation - Software cost, schedule, and risk estimation model. (Air Force wide license)
SEER-SSM	MS-DOS	Galorath Associates, Inc. Los Angeles, CA 90009 310-670-3404	Software size estimation - Produces software size estimates in lines of code or function points.
SIZE PLANNER	MS-DOS	Quantitative Software Mgmt., Inc. McLean, VA 22102 703-790-0055	Software size estimation - Uses four approaches for size estimation including fuzzy logic, function point, standard component, and new/reused/modified sizing.
SIZE Plus	X-Windows OSF,Motif, Unix, VMS	GEC-Marconi Software Systems Reston, VA 22090 703-648-1551	Software size estimation - Uses Function Point Analysis methodology.
SIZEEXPERT	MS-DOS	Technology Appl./Engr. Corp. Bethesda, MD 20817 301-571-8510	Software size estimation - Produces an estimate of lines of code based on cost expert questions.
SLIM	MS-DOS	Quantitative Software Mgmt., Inc. McLean, VA 22102 703-790-0055	Software cost estimation - Cost and schedule estimation with analytic tools for planning, control, risk, analysis. Uses expert system methodology.
SOFTCOST-Ada	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software cost estimation - One in a family of models for estimation of Ada projects.
SOFTCOST-R	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software cost estimation - One in a family of models for estimation of general projects.
SPQR/20	MS-DOS	Software Products Research Inc. Burlington, MA 01803 617-273-0140	Software cost estimation - In addition to software efforts, cost and schedule, this product estimates productivity, quality and reliability.
SWAN	MS-DOS	IIT Research Institute Rome, NY 13440-6916 315-336-2359	Software cost estimation - Estimates software development projects using COCOMO and size estimates with Function Point Analysis. Developed for US Army Program Manager for training devices (PMTRADE).
SYSTEM-4	MS-DOS	Computer Economics, Inc. Marina Del Rey, CA 90292 310-827-7300	Software cost estimation - Utilizes CEI's own basic estimating model.

Software Estimation Products List - By Vendor

Product	Platform	Vendor	Classification/Comments
REVIC	MS-DOS	Air Force Cost Analysis Agency Arlington, VA 22202 703-746-5865	Software cost estimation - Uses Intermediate COCOMO with added features for life cycle costing and risk analysis. (Public domain software)
SASET	MS-DOS	Air Force Cost Analysis Agency Arlington, VA 22202 703-746-5865	Software cost estimation - A forward chaining, rule-based expert system utilizing a hierarchically structured knowledge database.
COCOMOID	MS-DOS	Air Force Cost Center Wright-Patterson AFB, OH 45433 513-257-3927	Software cost estimation - Based on COCOMO. Provides estimates also based on enhanced, Ada, Ada process and incremental development models.
PROJECT BRIDGE	MS-DOS	Applied Business Technology Corp. New York, NY 10013-3992 800-444-0724	Software estimation - Estimates based on Function Point Analysis.
CA-ESTIMACS	MS-DOS	Computer Associates Int., Inc. Calverton, MD 20705 301-937-1133	Software cost estimation - Estimates efforts, schedule and cost of Information Systems software.
CA-FPXpert	MS-DOS	Computer Associates Int., Inc. Calverton, MD 20705 301-937-1133	Software size estimation - Uses Function Point Analysis methodology for size estimation.
CEIS	MS-DOS	Computer Economics, Inc. Marina Del Rey, CA 90292 310-827-7300	Software size estimation - Calculates four independent size estimates based on comparison to other known software projects.
SYSTEM-4	MS-DOS	Computer Economics, Inc. Marina Del Rey, CA 90292 310-827-7300	Software cost estimation - Utilizes CEI's own basic estimating model.
CB COCOMO	Mac/Windows	Decisioneering, Inc. Boulder, CO 80301 303-292-2291	Software cost estimation - Based on COCOMO to estimate effort and cost of software development projects.
SEER-SEM	MS-DOS	Galorath Associates, Inc. Los Angeles, CA 90009 310-670-3404	Software cost estimation - Software cost, schedule, and risk estimation model. (Air Force wide license)
SEER-SSM	MS-DOS	Galorath Associates, Inc. Los Angeles, CA 90009 310-670-3404	Software size estimation - Produces software size estimates in lines of code or function points.
PRICE S	MS-DOS	GE Price Systems Moorestown, NJ 08057 800-437-7423	Software cost estimation - One in a family of cost models for software and hardware costs, effort, and schedule. Uses functionality and source lines of code.
GECOMO Plus	X-Windows OSF,Motif, Unix, VMS	GEC-Marconi Software Systems Reston, VA 22090 703-648-1551	Software cost estimation - Implements an extended detailed COCOMO model and includes Ada.
SIZE Plus	X-Windows OSF,Motif, Unix, VMS	GEC-Marconi Software Systems Reston, VA 22090 703-648-1551	Software size estimation - Uses Function Point Analysis methodology.
GHL COCOMO	MS-DOS	GHL Associates, Inc. Haverford, PA 19041 215-896-7307	Software cost estimation - Estimates software development costs based on COCOMO model.
CoCoPro	Macintosh	Iconix Software Engineering, Inc. Santa Monica, CA 90405 310-458-0092	Software cost estimation - Estimates resources needed using standard COCOMO.
SECOMO	MS-DOS, VMS	IIT Research Institute Rome, NY 13440 315-336-2359	Software cost estimation - Uses full COCOMO maintenance cost estimation.
SWAN	MS-DOS	IIT Research Institute Rome, NY 13440-6916 315-336-2359	Software cost estimation - Estimates software development projects using COCOMO and size estimates with Function Point Analysis. Developed for US Army Program Manager for training devices. (PMTRADE)

Software Estimation Products List - By Vendor

Product	Platform	Vendor	Classification/Comments
COSTMODL	MS-DOS	NASA/JSC's Software Tech. Branch Houston, TX 77058 713-483-9092	Software cost estimation - Detailed COCOMO model. Available to government by NASA/JSC.
Micro Man ESTI-MATE	MS-DOS	POC-IT Management Services, Inc. Santa Monica, CA 90401 310-393-4552	Software cost estimation - Estimates effort for Information Systems projects using Function Point Analysis.
SIZE PLANNER	MS-DOS	Quantitative Software Mgmt., Inc. McLean, VA 22102 703-790-0055	Software size estimation - Uses four approaches for size estimation including fuzzy logic, function point, standard component, and new/reused/modified sizing.
SLIM	MS-DOS	Quantitative Software Mgmt., Inc. McLean, VA 22102 703-790-0055	Software cost estimation - Cost and schedule estimation with analytic tools for planning, control, risk, analysis. Uses expert system methodology.
ASSET-R	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software size estimation - One in a family of models for estimation of software development. Uses Function Point Analysis.
SOFTCOST-Ada	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software cost estimation - One in a family of models for estimation of Ada projects.
SOFTCOST-R	MS-DOS	Reifer Consultants, Inc. Torrance, CA 90510 310-373-8728	Software cost estimation - One in a family of models for estimation of general projects.
COSTAR	MS-DOS/VMS	Softstar Systems Amherst, NH 03031 603-672-0987	Software cost estimation - Uses detailed COCOMO, Ada COCOMO and sizing with function points.
CHECKPOINT	MS-DOS	Software Products Research , Inc. Burlington, MA 01803 617-273-0140	Software cost estimation - Knowledge-based software estimation tool. Uses function points and feature points to estimate size.
SPQR/20	MS-DOS	Software Products Research Inc. Burlington, MA 01803 617-273-0140	Software cost estimation - In addition to software efforts, cost and schedule, this product estimates productivity, quality and reliability.
COCOMO1	MS-DOS	Specs Inc. Junction City, OR 97448 503-998-8729	Software cost estimation - Artificial Intelligence Front End with COCOMO model.
COSTEXPERT	MS-DOS	Technology Appl./Engr. Corp. Bethesda, MD 20817 301-571-8510	Software cost estimation - "Expert System" based model. Does not use lines of code estimate or function points.
SIZEEXPERT	MS-DOS	Technology Appl./Engr. Corp. Bethesda, MD 20817 301-571-8510	Software size estimation - Produces an estimate of lines of code based on cost expert questions.

This page intentionally left blank

Appendix B

Software Estimation Product Sheets

PRODUCT SHEET - REPORT

Tool: **ASSET-R**

Version Number:	V2.2
Release Date:	3/92
Frequency of Updates:	Annually
Date of First Release:	3/87
Number Sold:	>60

Vendor: **Reifer Consultants, Inc.**

In Business Since: 1980

Address: P. O. Box 4046
Torrance, CA 90510

Point of Contact: Donald J. Reifer
Phone Number: 310-373-8728
Fax Number: 310-375-9845

Email Address:

Bulletin Board System:
Customer Support: 310-373-8728

Pricing:

Single User Price:	$8,000
Site License:	$11,000
Multi-Copy Price:	Negotiable
GSA Price:	
Maintenance Price:	$2,000 Annually

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

ASSET-R is a function point sizing tool developed to size real-time and scientific software systems directly from a requirements specification. It is a knowledge-based system which provides analyst support for implementing a sizing process which can be used early in the development cycle. Recent extensions have been made to deal with object-oriented specifications.

PRODUCT SHEET - REPORT

Tool: **CA-ESTIMACS**
Version Number: 7.0
Release Date: 1/92
Frequency of Updates: 12-18 months
Date of First Release: 1983
Number Sold: 5,000

Vendor: **Computer Associates International, Inc.**

In Business Since: 1977

Address: 4041 Powder Mill Rd., Suite 302
Calverton, MD 20705

Pricing:
Single User Price: $15,015 (GSA)
Site License: $43,355 (GSA)
Multi-Copy Price:

GSA Price:
Maintenance Price: 13% of purchase price (after first year)

Point of Contact: George Trainer
Phone Number: 301-937-1133
Fax Number: 301-595-7069

Email Address:

Bulletin Board System:
Customer Support: 800-645-3042

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

CA-ESTIMACS is an estimating model using research drawn from a data base of more than 14,000 completed software projects. CA-ESTIMACS is designed to provide software developers with time/cost/resource estimates (within ± 15% vs. actuals) during the feasibility phase of the life cycle. CA-ESTIMACS delivers estimates without requiring the user to know or understand function points and/or SLOC.

PRODUCT SHEET - REPORT

Tool: **CA-FPXpert**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Computer Associates International, Inc.**

In Business Since: 1977

Address: 4041 Powder Mill Rd., Suite 302
Calverton, MD 20705

Point of Contact: George Trainer
Phone Number: 301-937-1133
Fax Number: 301-595-7069

Email Address:

Bulletin Board System:
Customer Support: 800-645-3042

Pricing:
Single User Price: $9,870
Site License: $54,600
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

CA-FPXpert is a software size estimation tool that uses the FPA method for size estimation of Information Systems (IS) type software projects. It includes an on-line tutor to help with the function point counting process. Conforms to the standard counting practices published by the International Function Point User Group (IFPUG).

PRODUCT SHEET - REPORT

Tool: **CB COCOMO**
Version Number: 1.0
Release Date: 5/16/92
Frequency of Updates: Annually
Date of First Release: 1/1/91
Number Sold: 1,000

Vendor: **Decisioneering, Inc.**

In Business Since:

Address: 1724 Conestoga St.
Boulder, CO 80301

Point of Contact: Catherine LeRei
Phone Number: 303-292-2291
Fax Number: 303-292-9352

Email Address:

Bulletin Board System:
Customer Support: 303-447-6464

Pricing:
Single User Price: $295
Site License: Yes
Multi-Copy Price: Yes

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

Macintosh, Windows

Description/Purpose:

CB COCOMO is a software cost estimation tool based on COCOMO. It estimates time and cost of software development projects. It allows entering of actual project data at various phases in the life cycle to support cost to complete estimating. It is designed to work with Crystal Ball, a forecasting and risk analysis program. It delivers a range of estimates that shows the best case, worst case and most likely scenarios for schedules and budgets.

PRODUCT SHEET - REPORT

Tool: **CEIS**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Computer Economics, Inc.**

In Business Since: 1981

Address: 4560 Admiralty Way
Marina Del Rey, CA 90292-5424

Point of Contact: Tracy Thorpe
Phone Number: 310-827-7300
Fax Number: 619-632-0694

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $495
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

CEIS is a software size estimation tool. The estimation is based on information from other known software projects. This is accomplished by comparing the attributes (parameters) of the new project to the attributes of three reference projects of known size. Separate algorithms are used to produce four independent estimates and a level of confidence.

PRODUCT SHEET - REPORT

Tool: **CHECKPOINT**
Version Number: 2.1.5
Release Date: 4/92
Frequency of Updates: Annually
Date of First Release: 5/89
Number Sold: 115

Vendor: **Software Products Research, Inc.**

In Business Since: 1985

Address: 77 S. Bedford St.
Burlington, MA 01803

Point of Contact: John Zimmerman
Phone Number: 617-273-0140
Fax Number: 617-273-5176

Email Address:

Bulletin Board System:
Customer Support: 617-273-0140

Pricing:
Single User Price: $20,000
Site License: $99,000
Multi-Copy Price: Varies

GSA Price: $13,000
Maintenance Price: 10% Annually

Platforms/Operating Systems:

MS-DOS, UNIX (12/92)

Description/Purpose:

CHECKPOINT is a knowledge-based software measurement, estimation, and assessment tool that provides guidance and support for software managers and IS executives. It contains its own knowledge base of industry standards comprised of more than 4200 software projects from systems, IS and military environments, representing new projects, enhancements, and maintenance programs.

PRODUCT SHEET - REPORT

Tool: **COCOMO1**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Specs Inc.**

In Business Since:

Address: 95094 Turnbow Lane
Junction City, OR 97448

Point of Contact: Phil Chatman
Phone Number: 503-998-8729
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $99
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

COCOMO1 is a software cost estimation tool that uses COCOMO with an artificial intelligence front end. This is an expert system-based software tool. Through a series of questions, COCOMO1 assists in determining the proper COCOMO model and mode. COCOMO1 uses fifteen development and maintenance cost drivers and applies the formulas of COCOMO to these factors.

PRODUCT SHEET - REPORT

Tool: COCOMOID
Version Number: 3.183
Release Date: 12/91
Frequency of Updates: As required
Date of First Release: 8/86
Number Sold:

Vendor: Air Force Cost Center

In Business Since:

Address: AFMC/FMC
Wright-Patterson AFB, OH 45433-5001

Point of Contact: Ronnie E. Cooper
Phone Number: 513-257-3927
Fax Number: 513-476-1492

Email Address: rcooper@logdisl,hg,aflc,af,mil

Bulletin Board System:
Customer Support:

Pricing: Distributed free of license fee
Single User Price:
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

Lotus 1-2-3, version 2.01 up, WK1 format

Description/Purpose:

This is a complete COCOMO implementation including models for basic, intermediate, detailed, maintenance and calibration. It also includes enhanced, Ada, Ada Process model, and Incremental Development models.

PRODUCT SHEET - REPORT

Tool: **CoCoPro**
- Version Number: 2.0
- Release Date: 1988
- Frequency of Updates: Annually
- Date of First Release: 1986
- Number Sold: >200

Vendor: **Iconix Software Engineering, Inc.**

In Business Since: 1986

Address: 2800 28th St., Suite 320
Santa Monica, CA 90405

Point of Contact: Doug Rosenberg
- Phone Number: 818-708-0942
- Fax Number: 310-396-3454

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
- Single User Price: $995
- Site License: Negotiable
- Multi-Copy Price: 5-15% discount
- GSA Price: $845
- Maintenance Price: 15% of purchase price

Platforms/Operating Systems:

Macintosh

Description/Purpose:

Software development cost estimation tool based on Dr. Barry Boehm's COCOMO. Uses exponential functions and attributes to calculate development costs. Covers personnel experience and capabilities, project complexity, product factors and hardware limitations.

PRODUCT SHEET - REPORT

Tool: **COSTAR**
Version Number: 3.00
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Softstar Systems**

In Business Since:

Address: 28 Ponemah Rd.
Amherst, NH 03031

Point of Contact: Dan Ligett
Phone Number: 603-672-0987
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $800
Site License: $3,500
Multi-Copy Price: $400 for 3 to 7 copies

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS, VMS

Description/Purpose:

COSTAR is a full implementation of the detailed COCOMO model. It includes support for Ada COCOMO. Also, it utilizes function points for size estimation. It offers side-by-side comparison of several alternative estimates, automatic recalculation and display of results, and user definable cost drivers.

PRODUCT SHEET - REPORT

Tool: **COSTEXPERT/SIZEEXPERT**
Version Number: 2.06
Release Date: 3/11/91
Frequency of Updates:
Date of First Release: 7/1/88
Number Sold: 1

Vendor: **Technology Applications/ Engineering Corp.**

In Business Since: 1986

Address: 6701 Rockledge Dr., #330
Bethesda, MD 20817-1813

Point of Contact: Carl Krebs
Phone Number: 301-571-8510
Fax Number: 301-571-8513

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $2,500 Annually
Site License: Varies
Multi-Copy Price: Varies

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

COSTEXPERT is an expert system which estimates software development costs. The model is not based on lines of code or function points, but asks questions about the functionality of the software being developed. It directly estimates software related efforts such as program management, computer security engineering, and configuration management. It supports multiple languages and different development standards. SIZEEXPERT produces an estimate of lines of code based on the COSTEXPERT questions. These are two different tools, however, they are packaged and sold together.

PRODUCT SHEET - REPORT

Tool: **COSTMODL**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **NASA/JSC's Software Technology Branch**

In Business Since:

Address: 16511 Space Center Blvd.
Houston, TX 77058

Point of Contact: Bernie Roush
Phone Number: 713-483-9092
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price:
Site License:
Multi-Copy Price:

GSA Price: Government owned
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

COSTMODL is a tool for estimating the effort, cost, and schedule required to develop software products. It was developed by the Software Technology Branch of the Spacecraft Software Division at NASA/Johnson Space Center. Five different models are included for estimating non-Ada, Ada and products which are to be delivered as a series of incremental deliveries. The basic estimating equations can be calibrated to the user's software development environment type of products and the set of factors which influence software development.

PRODUCT SHEET - REPORT

Tool: **GECOMO Plus**
Version Number: 1.3
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **GEC-Marconi Software Systems**

In Business Since:

Address: 12110 Sunset Hill Rd., Suite 450
Reston, VA 22090

Point of Contact: Paul Raymond
Phone Number: 703-648-1551
Fax Number: 703-476-8035

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $4,000
Site License: $11,000 unlimited
Multi-Copy Price: $6,000 up to 5 users

GSA Price:
Maintenance Price: 15% of list price

Platforms/Operating Systems:

VMS, Unix OSF Motif, Windows

Description/Purpose:

GECOMO Plus is a cost estimating tool for software engineering projects. It is an enhancement of COCOMO. This product takes 17 cost drivers into account. It has implemented the Ada cost model allowing cost estimation for Ada-based projects as well as non-Ada projects.

PRODUCT SHEET - REPORT

Tool: **GHL COCOMO**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **GHL Associates, Inc.**

In Business Since:

Address: Haverford, PA 19041

Point of Contact: Winston Grisoff
Phone Number: 215-896-7307
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price:
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

COCOMO based cost estimation model. Features 3 levels of detail, multi-project structure, data retention, sensitivity analysis and what-ifs.

PRODUCT SHEET - REPORT

Tool: **Micro Man ESTI-MATE**
Version Number: 1.0
Release Date: 7/92
Frequency of Updates: Quarterly
Date of First Release:
Number Sold:

Vendor: **POC-IT Management Services, Inc.**
In Business Since: 1984

Address: 429 Santa Monica Blvd.
Suite 460
Santa Monica, CA 90401

Point of Contact: Janet Wachs
Phone Number: 310-393-4552; x214
Fax Number: 310-451-2888

Email Address: Compuserve 75766,1150

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $5,000
Site License: Yes
Multi-Copy Price: Yes

GSA Price:
Maintenance Price: 15% of purchase price

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

Micro Man ESTI-MATE is a PC-based estimating and planning tool for information services projects. It is used to calculate overall project effort and create project plans which include estimated hours for all project events. It uses FPA. The estimation is independent of programming language and the development environment.

PRODUCT SHEET - REPORT

Tool:	**PRICE S**	(Windows, Unix)
Version Number:		1.1
Release Date:		3/93
Frequency of Updates:		As required
Date of First Release:		1977
Number Sold:		

Pricing:	US Gov't	Commercial
Single User Price:	$1,000	$15,000-$17,000
Site License:		$32,500-$36,500
Multi-Copy Price:	Call	
LAN:		$37,500-$41,700
GSA Price:		
Maintenance Price:		

Vendor: **GE Price Systems**

In Business Since: 1975

Address: 300 Rt. 38, Bldg. 146
Moorestown, NJ 08057

Point of Contact: Frank Gray
Phone Number: 800-437-7423
Fax Number: 609-866-8889

Email Address:

Bulletin Board System:
Customer Support: 800-437-7423

Platforms/Operating Systems:

PRICE S - The model operates on workstations with Unix/Motif and PCs with Microsoft Windows.

Description/Purpose:

The PRICE S software model is a full featured software estimating tool. The tool estimates software size along with costs and schedule for the total software life cycle - from the system concept phase through maintenance and support. Output costs are provided in terms of effort (hours or months). It incorporates the effects of modern software development practices; the availability and use of software development tools; the experience of the software team; and the programming languages employed. PRICE S contains a graphical user interface, function point sizing and risk analysis. The windows product provides interface capability with Microsoft Excel and Project.

PRODUCT SHEET - REPORT

Tool: **PROJECT BRIDGE**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Applied Business Technology Corp.**
In Business Since:

Address: 361 Broadway
New York, NY 10013-3992

Point of Contact: Marie Mattox
Phone Number: 800-444-0724
Fax Number: 800-444-0726

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $5,000
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

PROJECT BRIDGE planning and estimating system is a knowledge-based tool for profiling, estimating and planning projects in a software engineering environment. It allows users to produce estimates based on function points or the organizations own estimating factors.

PRODUCT SHEET - REPORT

Tool: **REVIC**
Version Number: 9.11b
Release Date: 4/92
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Air Force Cost Analysis Agency**

In Business Since:

Address: 1111 Jefferson Davis Hwy., Suite 403
Arlington, VA 22202

Point of Contact: John B. Donald
Phone Number: 703-746-5865
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price:
Site License:
Multi-Copy Price:

GSA Price: Government owned
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

REVIC (Revised Intermediate COCOMO) contains a model for estimating the cost, schedule and manpower resources for software development projects. It is based on the Intermediate COCOMO model. In addition to providing estimates on cost, manpower, and schedule, the program creates estimates for typical DOD-STD-2167A documentation sizing and long term software maintenance.

PRODUCT SHEET - REPORT

Tool: **SASET**
Version Number:
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Air Force Cost Analysis Center**

In Business Since:

Address: 1111 Jefferson Davis Hwy., Suite 403
Arlington, VA 22202

Point of Contact: John B. Donald
Phone Number: 703-746-5865
Fax Number:

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price:
Site License:
Multi-Copy Price:

GSA Price: Government owned
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

Software Architecture Sizing and Estimating Tool (SASET) is a software cost estimation tool developed by Martin Marietta Corporation for the Naval Center for Cost Analysis. SASET estimates size, development effort and support for software development. It is a forward chaining, rule-based expert system utilizing a hierarchically structured knowledge data base. The data base is composed of projects with a wide range of applications.

PRODUCT SHEET - REPORT

Tool: SECOMO
Version Number: 7.0
Release Date: 11/89
Frequency of Updates:
Date of First Release: 9/85
Number Sold: 800+

Vendor: IIT Research Institute

In Business Since: 1957

Address: 201 Mill St.
Rome, NY 13440

Point of Contact: Anthony Williams
Phone Number: 315-336-2359
Fax Number: 315-339-7002

Email Address:

Bulletin Board System:
Customer Support:

Pricing: No Charge (need to provide diskettes)
Single User Price:
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

IBM PC or compatible/MS-DOS, VAX/VMS 3.2 or later

Description/Purpose:

The SECOMO interactive software cost estimation tool, based on COCOMO, calculates the total technical and support requirements for all phases of the software development cycle. Features include a user friendly interface, on-line help, and an expanded user's manual. Version 5.3 requires an IBM-PC or compatible running under MS-DOS with at least 384 kilobytes of memory.

PRODUCT SHEET - REPORT

Tool: **SEER-SEM**
Version Number: 3.2
Release Date: 3/93
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Galorath Associates, Inc.**

In Business Since: 1979

Address: P. O. Box 90579
Los Angeles, CA 90009

Point of Contact: Kathleen Jones
Phone Number: 310-670-3404
Fax Number: 310-670-6481

Email Address:

Bulletin Board System:
Customer Support: 310-670-3404

Pricing:
Single User Price: $15,000
Site License: $19,500
Multi-Copy Price: Call

GSA Price:
Maintenance Price: None

Platforms/Operating Systems:

MS-DOS, Macintosh, SUN/Unix

Description/Purpose:

SEER-SEM is a software cost, schedule and risk estimation tool that addresses all DOD software standards and requirements. Ada, DOD-STD-2167A, DOD-STD-1703, security and other current software issues are specifically supported. Knowledge base developed from thousands of completed DOD projects are an integral part of the model.

PRODUCT SHEET - REPORT

Tool: **SEER-SSM**
Version Number: 2.61
Release Date: 3/93
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **Galorath Associates, Inc.**

In Business Since: 1979

Address: P. O. Box 90579
Los Angeles, CA 90009

Point of Contact: Kathleen Jones
Phone Number: 310-670-3404
Fax Number: 310-670-6481

Email Address:

Bulletin Board System:
Customer Support: 310-670-3404

Pricing:
Single User Price: $1,950
Site License: $3,300
Multi-Copy Price: Call

GSA Price:
Maintenance Price: None

Platforms/Operating Systems:

IBM or compatible running DOS

Description/Purpose:

SEER-SSM is a software size estimation tool. It produces software size estimates in lines of code or function points. Provides its own historical data base to save time in producing the size estimates.

PRODUCT SHEET - REPORT

Tool: **SIZE PLANNER**
Version Number: 1.1
Release Date: 1/91
Frequency of Updates: Annually
Date of First Release: 1987
Number Sold: 350

Vendor: **Quantitative Software Management, Inc.**

In Business Since: 1978

Address: 2000 Corporate Ridge, Suite 900
McLean, VA 22102

Pricing:
Single User Price:
Site License: $10,000 Annually
Multi-Copy Price:

GSA Price:
Maintenance Price:

Point of Contact: Doug Putnam
Phone Number: 703-790-0055
Fax Number: 703-749-3795

Email Address:

Bulletin Board System:
Customer Support: 703-790-0055

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

This software size estimation tool uses four independent approaches for size estimation including fuzzy logic, function points, standard component and new/reused/modified sizing. Each approach views the product from a unique perspective. The multiple perspective approach provides a crosscheck on the overall estimate which reduces the uncertainty in the size estimate.

PRODUCT SHEET - REPORT

Tool: **SIZE Plus**
Version Number: 1.2
Release Date:
Frequency of Updates:
Date of First Release:
Number Sold:

Vendor: **GEC-Marconi Software Systems**

In Business Since: 1977

Address: 12110 Sunset Hills Rd., Suite 450
Reston, VA 22090

Point of Contact: Paul Raymond
Phone Number: 703-648-1551
Fax Number: 703-476-8035

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $2,800
Site License: $7,300 unlimited
Multi-Copy Price: $4,000 up to 5 users

GSA Price:
Maintenance Price: 15% of list price

Platforms/Operating Systems:

VMS, Unix OSF Motif, X-Windows

Description/Purpose:

SIZE Plus is used to estimate the size of software projects using the FPA method. Supports both data processing and real-time applications.

PRODUCT SHEET - REPORT

Tool: **SLIM**
Version Number: 3.0
Release Date: 1/92
Frequency of Updates: Annually
Date of First Release: 1978
Number Sold: ± 3,000

Vendor: **Quantitative Software Management, Inc.**
In Business Since:

Address: 2000 Corporate Ridge, Suite 900
McLean, VA 22102

Point of Contact: Doug Putnam
Phone Number: 703-790-0055
Fax Number: 703-749-3795

Email Address:

Bulletin Board System:
Customer Support: 703-790-0055

Pricing:
Single User Price:
Site License: $25,000 Annually
Multi-Copy Price: $1,000 each add'l

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

Software Life Cycle Model (SLIM) is a software cost, schedule, risk and reliability estimation tool for planning, control and risk analysis. It uses expert system methodology, and can be customized to a specific organization through the use of historic data.

PRODUCT SHEET - REPORT

Tool: **SOFTCOST-Ada**

Version Number:	V2.3
Release Date:	3/92
Frequency of Updates:	Annually
Date of First Release:	3/87
Number Sold:	>80

Vendor: **Reifer Consultants, Inc.**

In Business Since: 1980

Address: P. O. Box 4946
Torrance, CA 90510

Point of Contact: Donald J. Reifer
Phone Number: 310-373-8728
Fax Number: 310-375-9845

Email Address:

Bulletin Board System:
Customer Support: 310-373-8728

Pricing:

Single User Price:	$8,000
Site License:	$11,000
Multi-Copy Price:	Negotiable
GSA Price:	
Maintenance Price:	$2,000 Annually

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

SOFTCOST-Ada is a cost estimation tool specifically calibrated to predict the costs of systems packaged using object-oriented techniques (both Ada and C++). It builds on the experience of over 50 firms who have completed in excess of 200 projects which delivered over 50 million lines of code.

PRODUCT SHEET - REPORT

Tool: **SOFTCOST-R**
Version Number: V8.2
Release Date: 3/92
Frequency of Updates: Annually
Date of First Release: 3/85
Number Sold: >100

Vendor: **Reifer Consultants, Inc.**

In Business Since: 1980

Address: P. O. Box 4046
Torrance, CA 90510

Point of Contact: Donald J. Reifer
Phone Number: 310-373-8728
Fax Number: 310-375-9845

Email Address:

Bulletin Board System:
Customer Support: 310-373-8728

Pricing:
Single User Price: $5,000
Site License: $8,000
Multi-Copy Price: Negotiable

GSA Price:
Maintenance Price: $1,250 Annually

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

SOFTCOST-R is a cost estimation model specifically configured to predict the costs of real-time and scientific systems. Its mathematical formulation was derived to support risk analysis and "what-if" calculations. The package builds on the experience of over 1,500 completed projects none of which is over 5 years old.

PRODUCT SHEET - REPORT

Tool: **SPQR/20**
Version Number: 1.3b
Release Date: 1990
Frequency of Updates:
Date of First Release: 10/85
Number Sold: >250

Vendor: **Software Products Research Inc.**

In Business Since: 1985

Address: 77 South Bedford Street
Burlington, MA 01803

Point of Contact: John Zimmerman
Phone Number: 617-273-0140
Fax Number: 617-273-5176

Email Address:

Bulletin Board System:
Customer Support: 617-273-0140

Pricing:
Single User Price: $5,000
Site License: $25,000
Multi-Copy Price: Variable Disc.

GSA Price: $3,250
Maintenance Price: 10% Annually

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

Software Productivity, Quality, and Reliability, Twenty Questions (SPQR/20) is a software estimation tool that provides software project estimates for time, cost, and resource requirements. The key elements forecast by this tool are captured in its name: software productivity, quality and reliability.

PRODUCT SHEET - REPORT

Tool: **SWAN**
Version Number: 1.4
Release Date: 5/92
Frequency of Updates: As needed
Date of First Release: 4/91
Number Sold:

Vendor: **IIT Research Institute**

In Business Since: 1957

Address: 201 Mill St.
Rome, NY 13440-6916

Point of Contact: Tony Williams
Phone Number: 315-336-2359
Fax Number: 315-339-7001

Email Address:

Bulletin Board System:
Customer Support:

Pricing: No charge (need to provide diskettes)
Single User Price:
Site License:
Multi-Copy Price:

GSA Price: Government owned
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

Software Analysis Cost Model (SWAN) was developed for the U.S. Army Program Manager for Training Devices (PMTRADE) in Ada programming language. SWAN supports the intermediate version of COCOMO, including Ada COCOMO. It includes the FPA technique to assist users in determining software size estimates.

PRODUCT SHEET - REPORT

Tool: **SYSTEM-4**
Version Number: 4.1
Release Date: 1991
Frequency of Updates: Varies
Date of First Release: 1982
Number Sold:

Vendor: **Computer Economics, Inc.**

In Business Since:

Address: 4560 Admiralty Way, Suite 109
Marina Del Rey, CA 90292-5424

Point of Contact: Tracy Thorpe
Phone Number: 310-827-7300
Fax Number: 619-632-0694

Email Address:

Bulletin Board System:
Customer Support:

Pricing:
Single User Price: $8,900 Annually (3 units)
Site License:
Multi-Copy Price:

GSA Price:
Maintenance Price:

Platforms/Operating Systems:

MS-DOS

Description/Purpose:

SYSTEM-4 incorporates all major cost and schedule drivers and has available 1 to 40 year period of life cycle cost. The algorithms, formulas, sensitivities and programs in SYSTEM-4 are CEI's own state-of-the-art basic estimating model. Produces estimates for all program phases and all language types.

This page intentionally left blank

Appendix C:

Software Estimation Product Critiques

Software Estimation Product Critique

Tool:	**CA-ESTIMACS**
Vendor:	Computer Associates Int., Inc.
Version:	7.0
Hardware platform:	PC and Logicraft
Operating system:	MS-DOS
Memory used:	640K
Disk space used:	21.4 MB
Enhancements:	

Critique Author:	
Main Duties:	Evaluate technology for the BUSINESS-Pharmaceuticals.
Years of software experience:	7+
Years of experience with this tool:	2
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Used during problem definition phase and at end of feasibility phase.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Modification of questions (customize product); project base for estimates.

Notable weakness(es) of the tool:

Somewhat out of date with technology.

Advice for potential buyers of this tool:

It was the best we could find. It has a place in our development/maintenance cycle.

Software Estimation Product Critique

Tool:	**CA-ESTIMACS**
Vendor:	Computer Associates Int., Inc.
Version:	7.0
Hardware platform:	IBM PS/2-55
Operating system:	MS-DOS
Memory used:	640K
Disk space used:	1.9MB
Enhancements:	

Critique Author:	
Main Duties:	Manage the development efforts for core product and custom enhancements.
Years of software experience:	18+
Years of experience with this tool:	2+
Last time tool was used:	Currently
Author considers self software:	Manager

Environment of Use:
Project description:

Development and custom enhancement projects of 2-9 months in length. We use a modified traditional life cycle development method.

Overall impression of this tool:	Good
Quality of vendor support:	Fair

Notable strength(s) of the tool:

Flexibility in phase file definitions; can customize to one's own methodology and lifecycle.

Notable weakness(es) of the tool:

Not fully integrated with PLANMACS (version 6.0); not fully integrated with TIMELINE (version 4.0); current version (7.0) not compatible with MicroMan-II software.

Advice for potential buyers of this tool:

Have a good development/maintenance methodology and lifecycle in place. Have good project management policies, procedures and standards in place.

Software Estimation Product Critique

Tool:	**CA-FPXpert**
Vendor:	Computer Associates Int., Inc.
Version:	2.0
Hardware platform:	PC/Novell Network
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Implement corporate IS measurement. Corporate wide TQM trainer/facilitator.
Years of software experience:	10
Years of experience with this tool:	1
Last time tool was used:	Currently
Author considers self software:	Manager

Environment of Use:
Project description:

All project types.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Has the most functionality on the market; function point analyzer built into data input; function point calculator to get quick estimations.

Notable weakness(es) of the tool:

Multiple entries of like function point components can be tedious.

Advice for potential buyers of this tool:

First understand how you will use function points, then evaluate the product.

Software Estimation Product Critique

Tool:	**CHECKPOINT**
Vendor:	Software Products Research, Inc.
Version:	
Hardware platform:	
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software estimation and tracking.
Years of software experience:	14
Years of experience with this tool:	1
Last time tool was used:	Currently
Author considers self software:	Programmer

Environment of Use:
Project description:

Management Information Systems VAX and IRM platforms, requirement definition through installation, 500-1500 FPS.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Effort and documentation estimates are broken down by activities and tasks. Future plans for templating to allow estimates to be based on our own historical data should be useful.

Notable weakness(es) of the tool:

Does not support the information engineering life cycle; tool seems to overestimate effort and documentation sizes if the chart of accounts, complexity, language level, etc. are not set correctly - difficult to determine correct parameters to use.

Advice for potential buyers of this tool:

None.

Software Estimation Product Critique

Tool:	**CHECKPOINT**
Vendor:	Software Products Research, Inc.
Version:	2.1
Hardware platform:	DYNA 386/PC Clone
Operating system:	MS-DOS 5.0
Memory used:	~ 580K
Disk space used:	~ 6 MB
Enhancements:	

Critique Author:

Main Duties:	Lead software engineer on various real-time weapons systems; software analysis, design, code, and management.
Years of software experience:	13
Years of experience with this tool:	1.5
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Embedded weapons control systems (under MIL-SPEC developments) used at all phases beginning with initial requirements phase.

Overall impression of this tool:	Excellent
Quality of vendor support:	Good

Notable strength(s) of the tool:

Uses function points for system size input (vs. lines of code) - these can usually be estimated earlier than lines of code; based on a data base of ACTUAL software projects and their results; accounts for an extensive set of "soft factors" in a given development environment.

Notable weakness(es) of the tool:

Mapping the "tasks" to be included in the estimate to the tasks used in DOD (under DOD-STD-2167A) is somewhat difficult; the number of military, embedded systems projects included in the statistical data base is unknown - is there enough to get best estimate?

Advice for potential buyers of this tool:

Very good tool - should definitely try it and compare to the other tools used to estimate projects. However, must learn FPA to make full use of tool. The SPR, Inc. workshop series to learn to apply tool is pretty much required for best results.

Software Estimation Product Critique

Tool:	**CHECKPOINT**
Vendor:	Software Products Research, Inc.
Version:	2.0.7
Hardware platform:	PC (286)
Operating system:	DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Cost estimation and systems modeling for Internal Revenue Service (IRS) projects.
Years of software experience:	2
Years of experience with this tool:	1
Last time tool was used:	Currently
Author considers self software:	Engr/Progr

Environment of Use:
Project description:

Hardware/software government project. PRE-RFP Award 3000-6000 Function Points.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Quick turn-around for cost estimates. Good output reporting format from laser printer. Easy to understand the inputs and outputs. Good ability to use either function points or lines of code. Very good customer support.

Notable weakness(es) of the tool:

User inputs data and gets out results. The equations are hidden from user.

Advice for potential buyers of this tool:

Good for commercial software estimation. Good for FPA.

Software Estimation Product Critique

Tool:	**CHECKPOINT**
Vendor:	Software Products Research, Inc.
Version:	2.0.7
Hardware platform:	IBM PC 286
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software resource estimates, training, management.
Years of software experience:	12
Years of experience with this tool:	
Last time tool was used:	3 months
Author considers self software:	Mgr/Engineer

Environment of Use:
Project description:

Information systems, support tool, development and maintenance phases.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Function point sizing used; quick estimate and detailed; flexible phase accounting; good customer support.

Notable weakness(es) of the tool:

Requires many input parameters. Need training and experience.

Advice for potential buyers of this tool:

Get training. Need 386 machine.

Software Estimation Product Critique

Tool:	**CHECKPOINT**
Vendor:	Software Products Research, Inc.
Version:	2.0.7
Hardware platform:	
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	
Years of software experience:	
Years of experience with this tool:	1
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

User friendly; allows for quick comparisons among alternatives; uses detailed cost accounting framework; compares against industry standard.

Notable weakness(es) of the tool:

Requires estimates and assessments of technology and productivity; based on traditional software development life cycle; does not directly support information engineering approach.

Advice for potential buyers of this tool:

Tool is easy to use and can be calibrated in detail to model many software engineering applications.

Software Estimation Product Critique

Tool:	**COSTEXPERT**
Vendor:	Technology Appl./Engr. Corp.
Version:	
Hardware platform:	PC
Operating system:	DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software cost research and systems cost estimating.
Years of software experience:	14
Years of experience with this tool:	4
Last time tool was used:	>1 year
Author considers self software:	Engineer

Environment of Use:
Project description:

Type - C^3I; size - any; methods - waterfall/2167A.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Does not use lines of code/function points to estimate. Can be used with only answering 3 questions.

Notable weakness(es) of the tool:

Slow, unless you have a 486. Could use more types of outputs.

Advice for potential buyers of this tool:

This is a tool of the future that is probably slightly ahead of its time.

Software Estimation Product Critique

Tool:	**COSTAR**
Vendor:	Softstar Systems
Version:	3.0
Hardware platform:	PC 386
Operating system:	DOS 3.3
Memory used:	330K
Disk space used:	~ 1 MB
Enhancements:	

Critique Author:
Main Duties: Manage software department.

Years of software experience:	12
Years of experience with this tool:	3
Last time tool was used:	Currently
Author considers self software:	Manager

Environment of Use:
Project description:

Embedded applications ~ 15-20 KLOC per project; used during proposal preparation.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Supports report generation well; allows individual CSCs to be tested as children components to a CSCI level.

Notable weakness(es) of the tool:

Does not support performing an estimate based on a range of values (e.g., I would like to look at three possible scenarios of a nominal, best, and worst case and then create a single case based on a weighted distribution); does not support adding of DOD-STD-2167A documentation costs.

Advice for potential buyers of this tool:

Software Estimation Product Critique

Tool:	**COSTAR**
Vendor:	Softstar Systems
Version:	3.0
Hardware platform:	IBM-PC-AT
Operating system:	MS-DOS
Memory used:	330K
Disk space used:	1 MB
Enhancements:	

Critique Author:

Main Duties: Technical support of estimating system which includes development of software cost estimates.

Years of software experience:	15
Years of experience with this tool:	4
Last time tool was used:	Currently
Author considers self software:	Manager

Environment of Use:
Project description:

Tool is used during all project phases for all types of software developments.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Easy to use; integrated sizing (function point analysis) and cost estimation (COCOMO); very useful output reports; great support tool to support government audits.

Notable weakness(es) of the tool:

Probably the best COCOMO implementation available, however would be better if: schedules could be continuously adjusted with their ± 25% range; project level (all CSCIs) could be discretely combined and chronologically shifted.

Advice for potential buyers of this tool:

Read the documentation carefully - this tool has considerable capability to enhance its use and custom tailor solutions.

Software Estimation Product Critique

Tool:	**COSTMODL**
Vendor:	NASA/JSCs Software Technical Branch
Version:	5.2.1
Hardware platform:	PC
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Support proposal teams size cost estimation. Provide expert software engineering economics on the use of software cost models and their application.
Years of software experience:	24
Years of experience with this tool:	12
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Small to medium projects.

Overall impression of this tool:
Quality of vendor support:

Notable strength(s) of the tool:

Does include 1987-88 IOC Ada implementation; friendly user interface, and has an incremental development option.

Notable weakness(es) of the tool:

Should be used by persons familiar with the software process.

Advice for potential buyers of this tool:

Experience on software costing is a must.

Software Estimation Product Critique

Tool:	**GECOMO Plus**
Vendor:	GEC Marconi Software Systems
Version:	1.3
Hardware platform:	HP/APOLLO
Operating system:	UNIX SYSV and BSD
Memory used:	
Disk space used:	4 MB
Enhancements:	

Critique Author:	
Main Duties:	Software process evaluation, process definition, metrics and process improvement, tool introduction, estimation expert.
Years of software experience:	10
Years of experience with this tool:	2
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Real-time embedded radio communication software, 100 KLOC (mixed 'c' and ASM), SA/SD; used at all phases of the life cycle.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

X/Motif GUI; incremental estimation model; optimal staff override feature; networking capability; ability to build local historical data base; ability to mirror software system architecture in the estimation model; on-line help.

Notable weakness(es) of the tool:

Cannot calibrate to local conditions . . . no "easy" GUI method to calibrate effort and TDEV coefficients/exponents; cannot calibrate numerical effort/TDEV cost driver values; cannot add new cost drivers to model; cannot "tag" specific COCOMO historical projects or local historical projects to run estimate of current project.

Advice for potential buyers of this tool:

None.

Software Estimation Product Critique

Tool: **PRICE S**

Vendor: GE Price Systems

Version:
Hardware platform:
Operating system:
Memory used:
Disk space used:
Enhancements:

Critique Author:
Main Duties: Software resource estimates, training, management.

Years of software experience: 12
Years of experience with this tool: 11
Last time tool was used: Currently
Author considers self software: Mgr/Engineer

Environment of Use:
Project description:

MIS, avionics, C^3, support tools - development and maintenance.

Overall impression of this tool: Excellent
Quality of vendor support: Excellent

Notable strength(s) of the tool:

Good for military and commercial estimates; maps to DOD-STD 2167A; sensitivity analysis; can be calibrated; good vendor support; training available; cost estimator's tool.

Notable weakness(es) of the tool:

Uses time share on mainframe in New Jersey; not user friendly; expensive.

Advice for potential buyers of this tool:

Excellent tool for the sophisticated user; need training and experience; better for estimators than managers.

Software Estimation Product Critique

Tool:	**PRICE S**
Vendor:	GE Price Systems
Version:	
Hardware platform:	PC
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Cost analysis/acquisition support.
Years of software experience:	15
Years of experience with this tool:	10
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

The use of parametrics to develop software cost estimates very early in a program's life cycle. Also, its provisions for calibration to a known environment.

Notable weakness(es) of the tool:

The lack of visibility into the data base and algorithms used within the model.

Advice for potential buyers of this tool:

Do a lot of what-if sensitivity analyses to determine inter-relationships between variables.

Software Estimation Product Critique

Tool: **PRICE S**

Vendor: GE Price Systems

Version:
Hardware platform:
Operating system:
Memory used:
Disk space used:
Enhancements:

Critique Author:
Main Duties: Cost estimation and analysis, economic analysis of software development activity.

Years of software experience: 10
Years of experience with this tool: 8
Last time tool was used: Currently
Author considers self software: Engineer

Environment of Use:
Project description:

Real-time, C^3I, all sizes, all phases, maintenance.

Overall impression of this tool: Excellent
Quality of vendor support: Good

Notable strength(s) of the tool:

Real-time basis; radar data behind parametric equations; continuously updated.

Notable weakness(es) of the tool:

Use of dial-up will be much better with PC version; hard to use editor.

Advice for potential buyers of this tool:

Wait for PC version and get training.

Software Estimation Product Critique

Tool:	**PRICE S**
Vendor:	GE Price Systems
Version:	
Hardware platform:	
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software resource estimates - cost, schedule, risk analyses.
Years of software experience:	8
Years of experience with this tool:	5
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Avionics, military ground; phase: system concept through operations.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Mapped to DOD-STD-2167A; provides schedule durations by 2167A milestones; allows calibration for projects; PC version offers greater flexibility; can break software down to CSCIs and CSCs; GE Price support is good.

Notable weakness(es) of the tool:

Requires training; expensive.

Advice for potential buyers of this tool:

Make sure people participate in training (too many "goof-off" and do not learn how to use the model).

Software Estimation Product Critique

Tool: **REVIC**

Vendor: Air Force Cost Analysis Agency

Version: 9.0
Hardware platform: PC
Operating system: MS-DOS
Memory used:
Disk space used:
Enhancements:

Critique Author:
Main Duties: Support proposal teams size cost estimation. Provide expert software engineering economics on the use of software cost models and their application.

Years of software experience: 24
Years of experience with this tool: 12
Last time tool was used: Currently
Author considers self software: Engineer

Environment of Use:
Project description:

Used on various aerospace medium size projects.

Overall impression of this tool: Good
Quality of vendor support: Good

Notable strength(s) of the tool:

This tool has a good user interface for first time cost model users. It is an intermediate COCOMO version. Help is available from REVIC users group to calibrate tool to user's environment.

Notable weakness(es) of the tool:

Does not include IOC 1987-88 Ada coefficients and attributes; strictly Air Force dependent.

Advice for potential buyers of this tool:

If using on aerospace projects, one should understand the Ada implementation in the model thoroughly.

Software Estimation Product Critique

Tool:	**REVIC**
Vendor:	Air Force Cost Analysis Agency
Version:	Latest
Hardware platform:	PC
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	Per vendor requirements

Critique Author:
Main Duties: Support cost volume teams doing proposals which involve software cost estimation for development or maintenance.

Years of software experience:	37
Years of experience with this tool:	3
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Various aerospace proposals for all kinds of software estimates.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

This is an implementation of COCOMO but with a front end and back end for more phases in the life cycle. Has Graphical User Interface also.

Notable weakness(es) of the tool:

Does not use Boehm's coefficients as default; has non-standard implementation for Ada language estimates; user must know how to change coefficients.

Advice for potential buyers of this tool:

This tool is free, but it requires experience in the software cost estimation field to know and understand how best to use it.

Software Estimation Product Critique

Tool:	**SASET**
Vendor:	Air Force Cost Analysis Agency
Version:	2.03
Hardware platform:	IBM PC/compatible
Operating system:	DOS 2.1 +
Memory used:	640K
Disk space used:	1.5 MB
Enhancements:	Hard disk color monitor preferred

Critique Author:	
Main Duties:	Independent cost analysis of major Navy programs.
Years of software experience:	20
Years of experience with this tool:	5
Last time tool was used:	Currently
Author considers self software:	Mgr/Engr

Environment of Use:
Project description:

Life cycle cost estimation of software development and enhancement projects.

Overall impression of this tool:	Excellent
Quality of vendor support:	Good

Notable strength(s) of the tool:

Functional sizing data base; risk analysis of size, schedule and resource; calibration and date analysis capabilities.

Notable weakness(es) of the tool:

Training is available from the contractor only, and at a cost. User-friendliness features are currently being enhanced.

Advice for potential buyers of this tool:

Available at no cost to government organizations with a need to use.

Software Estimation Product Critique

Tool:	**SEER-SEM**
Vendor:	Galorath Associates, Inc.
Version:	
Hardware platform:	IBM PC compatible
Operating system:	DOS 3.0+
Memory used:	640K
Disk space used:	1 MB
Enhancements:	Color graphics +

Critique Author:	
Main Duties:	Independent cost analysis of major Navy program.
Years of software experience:	20
Years of experience with this tool:	5
Last time tool was used:	Currently
Author considers self software:	Mgr/Engr

Environment of Use:
Project description:

Life cycle cost estimation of software development and enhancement projects.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Uses friendly windows 3.1 interface; built-in knowledge bases for different software platforms; comparative analysis of different estimates with varying assumptions; risk evaluation capabilities.

Notable weakness(es) of the tool:

Somewhat difficult to calibrate.

Advice for potential buyers of this tool:

Recommend usage. One of the best tools on the market.

Software Estimation Product Critique

Tool:	**SEER-SEM**
Vendor:	Galorath Associates, Inc.
Version:	1.81
Hardware platform:	PC
Operating system:	MS-DOS
Memory used:	Vendor specs
Disk space used:	Vendor specs
Enhancements:	

Critique Author:
Main Duties: Support proposal teams size cost estimation. Provide expert software engineering economics on the use of software cost models and their application.

Years of software experience:	26
Years of experience with this tool:	12
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Example: C^3, 478,000 SLOC, waterfall development method. Projects of all sizes and knowledge bases. Used on large projects.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Excellent guide for management in evaluating software costs in proposals due to its excellent knowledge base.

Notable weakness(es) of the tool:

This tool should be used only by experienced software developers or software cost estimators. A non-knowledgeable person can get into serious trouble trying to interpret the I/O results.

Advice for potential buyers of this tool:

Be aware that non-technical software personnel should not attempt to use this tool.

Software Estimation Product Critique

Tool:	**SEER-SEM**
Vendor:	Galorath Associates, Inc.
Version:	Latest
Hardware platform:	PC
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	Per vendor requirements

Critique Author:	
Main Duties:	Support cost volume teams doing proposals which involve software cost estimation for development or maintenance.
Years of software experience:	37
Years of experience with this tool:	3
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Various aerospace types; various sizes; whatever RFP requires.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Handles all phases of software development and life cycle; produces good reports for cost estimation purposes.

Notable weakness(es) of the tool:

Requires training and thorough understanding to use.

Advice for potential buyers of this tool:

Tool is not for casual users; requires much training.

Software Estimation Product Critique

Tool:	**SLIM**
Vendor:	Quantitative Software Mgmt., Inc.
Version:	2.2
Hardware platform:	PC
Operating system:	DOS 5.0
Memory used:	
Disk space used:	1.3 MB
Enhancements:	

Critique Author:	
Main Duties:	Prepare and review software cost and schedule estimates, perform software project management training.
Years of software experience:	17
Years of experience with this tool:	7
Last time tool was used:	Currently
Author considers self software:	Mgr/Engineer

Environment of Use:
Project description:

Mostly command and control and real-time applications, size usually >10,000 SLOC, 2167A, all phases.

Overall impression of this tool:	Excellent
Quality of vendor support:	Excellent/Good

Notable strength(s) of the tool:

Easy to calibrate to past and current performance; excellent "what if" analysis; good/excellent risk analysis; integrated tool set (PAD5, SIZE and SLIM-CONTROL).

Notable weakness(es) of the tool:

Function button interface for current version; expensive for "small" shops; requires training and experience to use well.

Advice for potential buyers of this tool:

If you can afford one tool, buy SLIM. Expect big user interface improvement with 3.0 version. Buy the new book by Larry Putnam, it will help you learn how best to use SLIM. Get training!

Software Estimation Product Critique

Tool:	**SLIM**
Vendor:	Quantitative Software Mgmt. , Inc.
Version:	2.2
Hardware platform:	HP Vectra
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Estimating and tracking project development and third party vendor activity. Responsible for estimating/tracking processes.
Years of software experience:	4
Years of experience with this tool:	3
Last time tool was used:	Currently
Author considers self software:	Programmer

Environment of Use:
Project description:

Business systems.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Allows consistency and a quantifiable way to assess productivity.

Notable weakness(es) of the tool:

Cannot estimate using your own company's actuals; algorithms are based on all the industry's data; milestones are fixed and estimate accordingly; there is not enough data on client server/UNIX CASE tool development work.

Advice for potential buyers of this tool:

For establishing an initial estimating program, tool is extremely beneficial.

Software Estimation Product Critique

Tool: SOFTCOST-Ada

Vendor: Reifer Consultants, Inc.

Version: 2.1
Hardware platform: IBM PC
Operating system:
Memory used:
Disk space used:
Enhancements:

Critique Author:
Main Duties: Software resource estimates, training, management.

Years of software experience: 12
Years of experience with this tool: 3
Last time tool was used: Currently
Author considers self software: Mgr../Engineer

Environment of Use:
Project description:

Overall impression of this tool: Excellent/Good
Quality of vendor support: Good

Notable strength(s) of the tool:

Easy to use.; good for quick estimate; follows 2167A activities; good risk analysis; good for Ada projects.

Notable weakness(es) of the tool:

Cost estimating in maintenance phase; 2167A activities costs and management, CM and QA are not in basic estimate.

Advice for potential buyers of this tool:

Good for managers. Reasonable cost.

Software Estimation Product Critique

Tool:	**SOFTCOST-Ada**
Vendor:	Reifer Consultants, Inc.
Version:	
Hardware platform:	
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:
Main Duties:

Years of software experience:	10
Years of experience with this tool:	3
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

All types, Ada projects.

Overall impression of this tool:	Excellent
Quality of vendor support:	Good

Notable strength(s) of the tool:

PC based/supported; based on large amounts of data from past projects; Ada specific.

Notable weakness(es) of the tool:

Hard to reach support staff (Don Reifer); maintenance costly.

Advice for potential buyers of this tool:

Get training.

Software Estimation Product Critique

Tool:	**SOFTCOST-Ada**
Vendor:	Reifer Consultants, Inc.
Version:	2.1
Hardware platform:	IBM PC Mode 2/50
Operating system:	DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Cost estimation; systems engineering.
Years of software experience:	10
Years of experience with this tool:	5
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Program start, parametric software estimation.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Good refreshment of data base - all Ada programs; risk assessment - manloading; easy to use - good manual explaining model.

Notable weakness(es) of the tool:

Not as good for conversion.

Advice for potential buyers of this tool:

None.

Software Estimation Product Critique

Tool:	**SOFTCOST-R**
Vendor:	Reifer Consultants, Inc.
Version:	8.2
Hardware platform:	IBM PC
Operating system:	
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software resource estimates, training, management.
Years of software experience:	12
Years of experience with this tool:	4
Last time tool was used:	Currently
Author considers self software:	Mgr./Engineer

Environment of Use:
Project description:

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

Easy to use; good for quick estimate.

Notable weakness(es) of the tool:

Cost estimating in maintenance phase; 2167A activities costs and management, CM, and QA are not in basic estimate.

Advice for potential buyers of this tool:

Good for managers; reasonable cost.

Software Estimation Product Critique

Tool:	**SOFTCOST-R**
Vendor:	Reifer Consultants, Inc.
Version:	8.2
Hardware platform:	IBM PC
Operating system:	DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Cost analysis economic evaluation.
Years of software experience:	9
Years of experience with this tool:	1 month
Last time tool was used:	6 months
Author considers self software:	Engr/Progr

Environment of Use:
Project description:

Environmental data base relatively small amount of software (40K SLOC) used before software was developed.

Overall impression of this tool:	Good
Quality of vendor support:	Unknown

Notable strength(s) of the tool:

Provides results instantly; easy to use/format.

Notable weakness(es) of the tool:

Some variables meanings are not clear.

Advice for potential buyers of this tool:

None.

Software Estimation Product Critique

Tool: SOFTCOST-R

Vendor: Reifer Consultants, Inc.

Version:
Hardware platform:
Operating system:
Memory used:
Disk space used:
Enhancements:

Critique Author:
Main Duties: System engineer.

Years of software experience: 10
Years of experience with this tool: 3
Last time tool was used: Currently
Author considers self software: Engineer

Environment of Use:
Project description:

All non-Ada projects.

Overall impression of this tool: Excellent
Quality of vendor support: Good

Notable strength(s) of the tool:

PC based/supported; based on large amounts of data from past projects.

Notable weakness(es) of the tool:

Hard to reach support staff (Don Reifer); maintenance costly.

Advice for potential buyers of this tool:

Get training.

Software Estimation Product Critique

Tool:	**SOFTCOST-R**
Vendor:	Reifer Consultants, Inc.
Version:	8.2
Hardware platform:	IBM PC PS 2/50
Operating system:	PC DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Cost estimation; systems engineering.
Years of software experience:	10
Years of experience with this tool:	5
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Software estimation, parametric, program start.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Good refreshment of data base; easy to use; manual explains model; risk assessment; manloading.

Notable weakness(es) of the tool:

Not as good for conversion.

Advice for potential buyers of this tool:

None.

Software Estimation Product Critique

Tool:	**SOFTCOST-R**
Vendor:	Reifer Consultants, Inc.
Version:	8.2
Hardware platform:	
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Software resource estimates cost, schedule, risk analyses.
Years of software experience:	8
Years of experience with this tool:	5
Last time tool was used:	6 months
Author considers self software:	Engineer

Environment of Use:
Project description:

Avionics, C^2, telecomm, scientific.

Overall impression of this tool:	Good
Quality of vendor support:	Excellent

Notable strength(s) of the tool:

Includes security requirements as a project factor; easy to use; very user friendly; compares estimate of COCOMO; includes reuse in model; what if gaming; resource allocation; produces confidence ratings.

Notable weakness(es) of the tool:

Requires some training.

Advice for potential buyers of this tool:

Keep in contact with vendor - they are always doing leading edge work.

Software Estimation Product Critique

Tool:	**SPQR/20**
Vendor:	Software Products Research, Inc.
Version:	1.3b
Hardware platform:	PC compatible
Operating system:	MS-DOS
Memory used:	
Disk space used:	
Enhancements:	

Critique Author:	
Main Duties:	Instructor - software estimation tools.
Years of software experience:	14
Years of experience with this tool:	2.5
Last time tool was used:	Currently
Author considers self software:	Engineer

Environment of Use:
Project description:

Educational copy. Instructs students in use of tool.

Overall impression of this tool:	Good
Quality of vendor support:	Good

Notable strength(s) of the tool:

It is easy to use and learn. Use of function point analysis is good for size estimation.

Notable weakness(es) of the tool:

Estimation of support costs are limited to 5 years.

Advice for potential buyers of this tool:

None.

This page intentionally left blank

Appendix D:

Software Estimation Product Recommended Readings

RECOMMENDED READINGS

[Albrecht-Gaffney 83]

Albrecht, A.J., and Gaffney, J.E. "Software Functions, Source Lines of Code, and Development Effort Prediction: A Software Science Validation," IEEE Transactions on Software Engineering, Volume SE-9, Number 6, November 1983.

[Bailey 86]

Bailey, Elizabeth K., et al. "A Descriptive Evaluation of Automated Software Cost Estimation Models," Institute for Defense Analysis, Alexandria, VA, October 1986.

[Behrens 83]

Behrens, Charles A. "Measuring the Productivity of Computer Systems Development Activities with Function Points," IEEE Transaction on Software Engineering, Volume SE-9, Number 6, November 1983.

[Blalock 88]

Blalock, Crystal D. "An Analysis of Schedule Determination in Software Program Development and Software Development Estimation Models," September 1988.

[Boehm 81]

Boehm, Dr. Barry W. "Software Engineering Economics," Prentice-Hall, Englewood Cliffs, NJ, 1981.

[Boehm 86]

Boehm, Dr. Barry W. "Understanding and Controlling Software Costs," Proceedings of the 1986 IFIP Conference, Dublin Ireland, North-Holland, 1986.

[Boehm 87]

Boehm, Dr. Barry W. and Walter Royce. "Ada COCOMO: TRW IOC Version Proceedings," Third COCOMO USER's Group Meeting CMU Software Engineering Institute, Pittsburgh, PA., November 4, 1987.

[Boehm 88]

Boehm, Dr. Barry W. "A Spiral Model of Software Development and Enhancement," Tutorial: Software Engineering Project Management, IEEE, Washington, DC, 1988.

[CEI 89]

Computer Economics, Incorporated. "System-4 User's Manual," Marina del Rey, CA, 1989.

[Conner 82]

Conner, Daryl R., and Robert W. Patterson. "Building Commitment to Organizational Change," *Training and Development Journal*, Vol. 36, Number 4, April 1982, pp. 18-30.

[Daly 90]

Daly, Bryan A.. "A Comparison of Software Schedule Estimators," (AFIT Thesis GCA/LSQ/905-1), Air Force Institue of Technology, Dayton, OH, 1990.

[Davis 88]

Davis, Alan M., Edward H. Bergoff, and Edward R. Comer. "A Strategy for Comparing Alternative Software Development Life Cycle Models," IEEE Transactions on Software Engineering, 14:1453-1461, October 1988.

[DOC 93]

Petersen, Gary. "Documentation Technology Report," Software Technology Support Center, March 1993.

[DOD 85]

Department of Defense. Technical Reviews and Audits for Systems, Equipment, and Computer Programs; Military Standard 1521B. Washington DOD, 4 June 1985.

[DOD 88]

Department of Defense. Defense System Development Military Standard 2167A. Washington DOD, 29 February 1988.

[Dreger 89]

Dreger, Brian J. "Function Point Analysis," Prentice-Hall, Inc., 1989.

[Ferens 86]

Ferens, Daniel V. "A Common Sense Approach to Software Cost Estimation," ISPA Journal of Parametrics, Volume VI, Number 2, June 1986.

[Ferens 1/91]

Ferens, Daniel V. "An Introduction To Software Cost Estimation and Measurement," January, 1991.

[Ferens 91]

Ferens, Daniel V. "Evaluation of Eight Software Support Cost Models," National Estimator, Spring 1991, pp 2-12.

[Ferens-Albanese 89]

Ferens, Daniel V. and Albanese, Frank, Jr. "Software Size Estimation: Deja Vu?" ISPA Journal of Parametrics, Volume IX, Number 3, October 1989.

[Ferens-Daly]

Ferens, Daniel V. and Daly, Bryan A. "A Quantitative Comparison of Popular Software Scheduling Models," ISPA 1991 Conference Proceedings, pp. SW-43-61.

[Fowler 88]

Fowler, Pricilla and Stan Przybylinski. "Transferring Software Engineering Tool Technology," IEEE Computer Society Press, Washington D.C., 1988.

[Freiman-Park 79]

Freiman, F. R. and Park, R.E. "Price Software Model - Version 3: An Overview," Proceedings, IEEE - Piny Workshop on Quantitative Software Models, IEEE Catalog No. TH0067-9, October 1979, pp. 32-41.

[Galorath 92]

Galorath Associates. "SEER User's Manual", Los Angeles, CA, 1989.

[Greathouse-Shipley 90]

Greathouse, Richard M. and Shipley, Kelly L. "Current Research on Schedulers for Aerospace Industry Software," Society of Automotive Engineers, Dayton, OH, 1990.

[GUIDE 85]

GUIDE International Corporation, "Estimating Using Function Points Handbook," (GPP-134), Chicago, IL, 1985.

[IITRI 89]

Illinois Institute of Technology (IIT) Research Institute. "Test Case Study: Estimating the Cost of Ada Software Development," Lanham, MD, April 1989.

[IITRI 91]

IIT Research Institute, "Swan Cost Model," Rome, NY, August 1991.

[ISA 90]

ISA Corporation. "ISA Cost Expert User's Guide," Technical Applications/Engineering Corp., Bethesda, MD, 1990.

[Jensen 81]

Jensen, R. W. "A Macro-level Software Development Cost Estimation Methodology," Proceedings of Fourteenth Asilomar Conference on Circuits, Systems and Computers, IEEE, New York, 1981.

[Jones 86]

Jones, Capers. "Programming Productivity," New York, McGraw-Hill, 1986.

[Jones 88]

Jones, Capers. *Computerworld*, p. 97, November 7, 1988.

[Jones 91]

Jones, Capers. "Applied Software Measurement; Assuring Productivity and Quality," McGraw-Hill, New York, NY, 1991.

[Kyle 91]

Kyle, Raymond A. "REVIC Software Cost Estimating Model User's Manual," Version 9.0, Albuquerque, NM, Air Force Contract Management Division, February 1991.

[NASA/JSC 91]

"COSTMODL User's Guide," NASA Johnson Space Center, Houston, TX, Version 5.2, January 1991.

[PM 92]

Steadman, Todd. "Project Management Tools Report," Software Technology Support Center, March 1992.

[Price 88]

Price Systems. "The Central Equations of the Price Software Cost Model," Proceedings of the Fourth Annual COCOMO User's Group Meeting, Pittsburg, PA, Software Engineering Institute, November 1988.

[Price 89]

GE Price Systems, "PRICE S User's Manual," Moorestown, NJ, Price Systems 1989.

[Putnam 92]

Putnam, L. and Myers, W. "Measures for Excellence: Reliable Software, On Time, Within Budget," Yourdon Press, 1992.

[Putnam-Fitzsimmons 79]

Putnam, L. H. and Fitzsimmons, A. "Estimating Software Costs," Datamation, September-November 1979.

[QSM-SIZE PLANNER 87]

QSM Corporation, "SIZE PLANNER User's Manual," Version 1.0, McLean, VA, QSM, 1987.

[QSM-SLIM 87]

QSM Corporation, "Slim User's Manual," McLean, VA, QSM: 1987.

[RAD 92]

Bowler, Odean. "Requirements Analysis and Design Tools Report," Software Technology Support Center, 1992.

[RCI 89]

Reifer, Donald J. "SOFTCOST-R User's Manual," Version 8.0, Torrance, CA, Reifer Consultants, Inc, 1989.

[RE 92]

Sittenauer, Chris. "Reengineering Tools Report," Software Technology Support Center, 1992.

[SCSA 92]

Price, Gordon. "Source Code Static Analysis Tools Report," Software Technology Support Center, 1992.

[SEE 92]

Hanrahan, Robert P. "Software Engineering Environments Report," Software Technology Support Center, 1992.

[Shere 88]

Shere, Kenneth D. "Software Engineering and Management," Englewood Cliffs, NJ, Prentice-Hall, 1988.

[Silver 90]

Silver, Dr. Aaron, and others. "SASET Users Guide," Publication R-0330-90-6, Naval Center for Cost Analysis, Department of the Navy, Washington DC, August 1990.

[SPR 86]

Software Productivity Research, Inc., "SPQR/20 User's Guide," Cambridge, MA, Software Productivity Research, 1986.

[Symons 91]

Symons, Charles R. "Software Sizing and Estimating," John Wiley and Sons, Inc., New York, NY, 1991.

[Tausworthe 81]

Tausworthe, R. C. "Deep Space Network Cost Estimation Model," Jet Propulsion Laboratory, JPL Publication 81-7, April 1981.

[TPEE 93]

Daich, Greg. "Test Preparation, Execution, and Evaluation," Software Technology Support Center, 1993.

Appendix E:

Software Estimation Products Glossary

Definitions

Algorithmic Models - (also known as Parametric models) produce a software cost estimate using one or more mathematical algorithms using a number of variables considered to be the major cost drivers. These models estimate effort or cost based primarily on the size of the software or Delivered Source Instruction (DSIs), and other productivity factors known as cost driver attributes.

Analogy Models - use a method of software estimation that involves comparing a proposed project with one or more similar completed projects where costs and schedules are known. Then, extrapolating from the actual costs of completed projects, estimate the cost of a proposed project.

Annual Change Traffic (ACT) - the fraction of the software product's source instructions which undergoes change during a year, either through addition or modification. The ACT is the quantity used to determine the product size for software maintenance effort estimation.

Basic COCOMO Model - estimates the effort required to develop software in three modes of development (Organic, Semidetached, or Embedded) using only DSIs as an input. The Basic model is good for quick, early, and rough order of magnitude estimates.

Bottom-Up Models - use a method of software estimation that estimates each component of the software project separately, and the results are combined to produce an estimate of the entire project.

COnstructive COst MOdel (COCOMO) - a software cost estimation model developed by Dr. Barry Boehm and is described in his book "Software Engineering Economics" (Boehm, 1981).

Cost/Schedule Control System Criteria (C/SCSC) - a set of criteria specified by the Federal Government for reporting project schedule and financial information.

Computer-Aided Software Engineering (CASE) - identifies a sector of the computer software industry concerned with producing software development environments and tools. The main components of a CASE product are individual tools to aid the software

developer or project manager during one or more phases of software development (or maintenance). Other features are a common user interface; interoperability of tools; and a repository or encyclopedia to provide a common tool base and central project database. CASE may also provide for code generation.

Configuration Item - hardware or software, or an aggregate of both, which is designated by the project configuration manager (or contracting agency) for configuration management.

Configuration Management - a discipline applying technical and administrative controls to (1) identification and documentation of physical and functional characteristics of configuration items; (2) any changes to characteristics of those configuration items; and (3) recording and reporting of change processing and implementation of the system.

Cost Driver Attributes - productivity factors in the software product development process that include software product attributes, computer attributes, personnel attributes, and project attributes.

Delivered Source Instructions (DSIs) - the number of source lines of code developed by the project. The number of DSIs is the primary input to many software cost estimating tools. The term **DELIVERED** is generally meant to exclude non-delivered support software such as test drivers. However, if these are developed with the same care as delivered software, with their own reviews, test plans, documentation, etc., then they should be counted. The term **SOURCE INSTRUCTIONS** includes all program instructions created by project personnel and processed into machine code by some combination of preprocessors, compilers, and assemblers. It excludes comments and unmodified utility software. It includes job control language, format statements, and data declarations.

Delphi Technique - a group forecasting technique, generally used for future events such as technological developments, that uses estimates from experts and feedback summaries of these estimates for additional estimates by these experts until reasonable consensus occurs. It has been used in various software cost-estimating activities, including estimation of factors influencing software costs.

Detailed COCOMO Model - differs from the Intermediate COCOMO model in that it uses effort multipliers for each phase of the project. These phase dependent effort multipliers yield better estimates because the cost driver ratings may be different during each phase. The detailed model also provides a three-level product hierarchy and has some other capabilities such as a procedure for adjusting the phase distribution of the development schedule.

DOD-STD-2167A - a U. S. Department of Defense standard that specifies the overall process for the development and documentation of mission-critical software systems.

Domain - a specific phase or area of the software life cycle in which a developer works. Domains define developers and users areas of responsibility and the scope of possible relationships between products. The work at the Software Technology Support Center is organized by domains such as Software Engineering Environments, Documentation, Project Management, etc.

Effort Adjustment Factor (EAF) - a term used in COCOMO to calculate the cost driver attribute's effect on the project. It is the product of the effort multipliers corresponding to each of the cost drivers for the project.

Embedded Mode - a term used by COCOMO to describe a project development that is characterized by tight, inflexible constraints and interface requirements. The product must operate within (is embedded in) a strongly coupled complex of hardware, software, regulations and operational procedures. An embedded mode project will require a great deal of innovation. An example would be a real-time system with timing constraints and customized hardware.

Expert Judgment Models - use a method of software estimation that is based on consultation with one or more experts that have experience with similar projects. An expert-consensus mechanism such as the **Delphi Technique** may be used to produce the estimate.

Intermediate COCOMO Model - an extension of the Basic COCOMO model. The Intermediate model uses an Effort Adjustment Factor (EAF) and slightly different coefficients for the effort equation than the Basic model. The Intermediate model produces better results than the Basic model because the user supplies settings for cost

drivers that determine the effort and duration of the software projects. The Intermediate model also allows the system to be divided and estimated in components. DSI values and cost drivers can be chosen for individual components instead of for the system as a whole.

Knowledge Base - the repository of knowledge in a computer system or organization. The collection of data, rules, and processes that are used to control a system, especially one using artificial intelligence or expert system methods.

Life Cycle - the stages and processes through which software passes during its development and operational use. Useful life of a system. Its length depends on the nature and volatility of the business, as well as the software development tools used to generate the databases and applications.

Management Information Systems - a computer based system of processing and organizing information so as to provide different levels of management within an organization with accurate and timely information needed for supervising activities, tracking progress, making decisions, and isolating and solving problems.

Metric - Quantitative analysis values calculated according to a precise definition and used to establish comparative aspects of development progress, quality assessment or choice of options.

Motif - Open Software Foundation (OSF/Motif) - graphical user interface from OSF that provides a Presentation Manager look and feel for applications running on any system with X Window Version 11. It conforms to POSIX, ANSI C, and X/Open's XPG3 standards.

Organic Mode - a term used by COCOMO to describe a project that is developed in a familiar, stable environment. The product is similar to previously developed products. Most people connected with the project have extensive experience in working with related systems and have a thorough understanding of the project. The project contains a minimum of innovative data processing architectures or algorithms. The product requires little innovation and is relatively small, rarely greater than 50,000 DSIs.

Paradigm - a model, example, or pattern. A generally accepted way of thinking.

Platform - hardware architecture of a particular model or family of computers. The term sometimes refers to the hardware and its operating system.

Procedures - manual procedures are human tasks. Machine procedures are lists of routines or programs to be executed, such as described by the job control language (JCL) in a mini or mainframe, or the batch processing language in a personal computer.

Process - the sequence of activities (in software development) described in terms of the user roles, user tasks, rules, events, work products, resource use, and the relationships between them. It may include the specific design methodology, language, documentation standards, etc.

Program Evaluation and Review Technique (PERT) - a method used to size a software product and calculate the Standard Deviation (SD) for risk assessment. The PERT equation (beta distribution) estimates the Equivalent Delivered Source Instructions (EDSIs) and the SD based on the analyst's estimates of the lowest possible size, the most likely size, and the highest possible size of each computer program component (CPC).

Rapid Prototyping - the creation of a working model of a software module to demonstrate the feasibility of the function. The prototype is later refined for inclusion in a final product.

Rayleigh Distribution - a curve that yields a good approximation to the actual labor curves on software projects.

Real-time - (1) Immediate response. The term may refer to fast transaction processing systems in business; however, it is normally used to refer to process control applications. For example, in avionics and space flight, real-time computers must respond instantly to signals sent to them.

(2) Any electronic operation that is performed in the same time frame as its real-world counterpart. For example, it takes a fast computer to simulate complex, solid models moving on screen at the same rate they move in the real world. Real-time video transmission produces a live broadcast.

Re-engineering - process of restructuring and redesigning an operational (or coded) software system to make it meet certain style, structure, or performance standards.

Reusability - ability to use all or the greater part of the same programming code or system design in another application.

Reuse - software development technique that allows the design and construction of reusable modules, objects, or units, that are stored in a library or database for future use in new applications. Reuse can be applied to any methodology in the construction phase, but is most effective when object oriented design methodologies are used.

Security - the protection from accidental or malicious access, use, modification, destruction, or disclosure. There are two aspects to security, confidentiality and integrity.

Semidetached Mode - a term used by COCOMO to describe a project that is developed somewhere between organic and embedded. The team members have a mixture of experienced and inexperienced personnel. The software to be developed has some characteristics of both organic and embedded modes. Semidetached software can be as large as 300K DSIs.

Software Development Life Cycle - the stages and processes through which software passes during its development. This includes requirements definition, analysis, design, coding, testing, and maintenance.

Software Development Life Cycle Methodology - application of methods, rules, and postulates to the software development process to establish completeness criteria, assure an efficient process, and develop a high quality product.

Software Method - (or Software Methodology) - focuses on how to navigate through each phase of the software process model (determining data, control, or uses hierarchies; partitioning functions; and allocating requirements) and how to represent phase products (structure charts; stimulus-response threads; and state transition diagrams).

Software Tool - program that aids in the development of other software programs. It may assist the programmer in the design, code, compile, link, edit, or debug phases.

Top-Down Models - use a method of software estimation that estimates the overall cost and effort of the proposed project derived from global properties of the project. The total cost and schedule is partitioned into components for planning purposes.

Work Breakdown Structure - a way of grouping the project's tasks and subtasks. The tasks are numbered to indicate their relationship to each other. Some industries use established work breakdown structure systems for billing and reporting purposes.

Workstation - high-performance, single user microcomputer or minicomputer that has been specialized for graphics, CAD, CAE, or scientific application.

Acronyms

ACAP	-	Analyst Capability
ACT	-	Annual Change Traffic
AEXP	-	Applications Experience
AFIT	-	Air Force Institute of Technology
AFLC	-	Air Force Logistics Command
AI	-	Artificial Intelligence
ALC	-	Air Logistics Center
ASCII	-	American Standard Code for Information Interchange
ASSCM	-	Avionics Software Support Cost Model
C/SCSC	-	Cost/Schedule Control System Criteria
CA	-	Computer Associates
CDR	-	Critical Design Review
CEI	-	Computer Economics Incorporated
CER	-	Cost Estimating Relationships
CF	-	Complexity Factor
COBOL	-	Common Business-Oriented Language
COCOMO	-	COnstruction COst MOdel
COSMIC	-	Computer Software Management and Information Center
CPI	-	Continuous Process Improvement
CPLX	-	Product Complexity
CSC	-	Computer Software Component
CSCI	-	Computer Software Configuration Item
CSU	-	Computer Software Unit
DATA	-	Database Size
DOD	-	Departement of Defense
DSI	-	Delivered Source Instructions
EAF	-	Effort Adjustment Factor
EDSI	-	Equivalent Delivered Source Instructions
EM	-	Effort Multiplier
FC	-	Function Count
FCA	-	Functional Configuration Audit
FP	-	Function Point
FPA	-	Function Point Analysis

FQR	-	Formal Qualification Review
GE	-	General Electric
GUI	-	Graphical User Interface
IFPUG	-	International Function Point Users Group
IIT	-	Illinois Institute of Technology
IITRI	-	Illinois Institute of Technology Research Institute
IOC	-	Initial Operational Capability
IS	-	Information System
ISA	-	Institute for Systems Analysis
JSC	-	Johnson Space Center
KISS	-	Keep-It-Simple Stupid
LEXP	-	Programming Language Experience
MBI	-	Manpower Build-up Index
MM	-	Man-Month
MODP	-	Modern Programming Practices
MS-DOS	-	Microsoft - Disk Operating System
NASA	-	National Aeronautics and Space Administration
NATO	-	North Atlantic Treaty Organization
OOA	-	Object-Oriented Analysis
OOD	-	Object-Oriented Design
PC	-	Personal Computer
PCA	-	Physical Configuration Audit
PCAP	-	Programmers Capability
PDR	-	Preliminary Design Review
PERT	-	Program Evaluation and Review Technique
PI	-	Productivity Index
PL/1	-	Programming Language 1
PMTRADE	-	Program Manager for Training Devices
QSM	-	Quantitative Software Management
RCI	-	Reifer Consultants Incorporated
RELY	-	Required Software Reliability
REVIC	-	Revised Enhanced Version of Intermediate Cocomo
RUG	-	REVIC User Group
RUSE	-	Required Reuseability
SASET	-	Software Architecture, Sizing and Estimating Tool
SCEA	-	Society for Cost and Economic Analysis

SCED	-	Required Development Schedule
SD	-	Standard Deviation
SDDD	-	Software Detailed Design Document
SDR	-	System Design Review
SECOMO	-	Software Engineering Cost Model
SECU	-	Security Classified Projects
SEER	-	System Evaluation and Estimation of Resources
SLIM	-	Software Life Cycle Model
SLOC	-	Source Lines of Code
SMG	-	Software Management Guide
SOW	-	Statement of Work
SPECS	-	Solar Powered Emergency Communication Systems Incorporated
SPQR/20	-	Software Productivity, Quality, and Reliability, Twenty Questions
SPR	-	Software Productivity Research
SRS	-	Software Requirements Specification
SSR	-	Software Specification Review
SSS	-	System Segment Specification
STLDD	-	Software Top-Level Design Document
STOR	-	Main Storage Constant
STSC	-	Software Technology Support Center
SWAN	-	Software Analysis Cost Model
TDEV	-	Total Development
TIME	-	Execution Time Constant
TOOL	-	Use of Software Tools
TRR	-	Test Readiness Review
TURN	-	Computer Turnaround Time
VEXP	-	Virtual Machine Experience
VIRT	-	Virtual Machine Volatility
VMVH	-	Virtual Host Machine
VMVT	-	Virtual Target Machine
WBS	-	Work Breakdown Structure

Standards

DOD-STD-2167A, Software Development Defense Systems, 1985

DOD-STD-1521B, Technical Reviews and Audits for Systems, Equipment, and Computer Software, 1985

MIL-STD-480B, Configuration Control-Engineering Changes, Deviations and Waivers

MIL-STD-499D, Engineering Management

DOD-STD-1703, Software Products Standard

Appendix F:

COCOMO

COCOMO

COCOMO was developed by Dr. Barry Boehm and a book was published in 1981. The complete COCOMO model and associated data base of 63 projects, which the model was developed from, appear in his book "Software Engineering Economics" [Boehm 81].

COCOMO uses Effort Equations to estimate man-months (MM) of effort required to complete a software project. Most of the other calculations are based on these effort equations, as well as project size estimates and other key cost drivers. Size estimations are defined in terms of Delivered Source Instructions (DSIs) and are defined by the user.

Cost drivers are defined in terms of product attributes, personnel attributes, computer attributes, and project attributes. For example, personnel attributes include analytical capability, programming capability, virtual machine experience, language experience, and application experience. Each attribute applies a multiplier which determines impact to overall estimates.

COCOMO considers project development to be done in one of three available modes. These modes are an attempt to classify and define three types of environments in which software products are developed. They are the Organic Mode, Semidetached Mode, and Embedded Mode.

Organic Mode - This mode is used when the project is being developed in a familiar, stable environment, and the product is similar to previously developed products. The project is also relatively small, requires little innovation, and is usually less than 50,000 DSIs. An example is an accounting system.

Embedded Mode - The project is characterized by tight, inflexible constraints and interface requirements, and requires a great deal of innovation. For example, real-time systems with critical timing constraints and customized hardware.

Semidetached Mode - Project characteristics are somewhere between those described for the Organic Mode and those of the Embedded Mode. Project size can be as large as 300K DSIs.

COCOMO is also defined in terms of three different models. The more complex models address more factors that influence software projects and are therefore more accurate. The three models are Basic Model, Intermediate Model, and the Detailed Model.

Basic Model - Estimates for required effort are based primarily on user estimations of project size. This model is more suitable for early, rough, estimates of a project's effort, duration, and cost. Accuracy is generally within a factor of 2 of the actual results 60% of the time.

Intermediate Model - This model uses an Effort Adjustment Factor (EAF) and slightly different coefficients for the effort equations than those used by the Basic Model. The EAF is computed from cost driver attributes. This model also produces better results than the Basic Model because DSI values and Cost Drivers can be assigned to a project's individual components. Estimation accuracy is usually within 20% of the actual results 68% of the time.

Detailed Model - This model differs from the Intermediate Model by using different Effort Multipliers for each phase of a project. Phases include Requirements, Product Design, Detailed Design, Code and Unit Test, Integrate and Test, and Maintenance.

COCOMO provides cost driver ratings for estimating project maintenance as well as projects developed using Ada. Also, the size of projects in the COCOMO data base range from around 2000 to 512,000 DSIs; therefore, estimates should not be considered reliable when DSIs are outside this range.

The COCOMO Maintenance model assumes that the software maintenance has about the same cost driver attributes as those that determine software development . The quantity used to determine the product size for software maintenance is the Annual Change Traffic (ACT). The ACT is the fraction of the software product's source instructions which undergo change during a year, either through addition or modification.

Many factors that influence the development of Ada software have not been considered by the standard COCOMO estimation equations. Therefore, an Initial Operational Capability (IOC) Ada COCOMO has been developed for estimating software

products developed in the Ada programming language. For more detailed information see Ada COCOMO: TRW IOC Version, [Boehm 87].

Appendix G:

Function Point Analysis

Function Point Analysis

The Function Point Analysis (FPA) method of software size estimating was developed by Allan J. Albrecht and John E. Gaffney, Jr. [Albrecht and Gaffney, 83] at IBM. This method is used primarily for estimating the size of Information System (IS) software development projects.

The objective of FPA is to determine a numeric rating or total Function Point Count (FPC) for the project by measuring logical functions the project is to perform and adjusting the FPC according to a Processing Complexity Adjustment (PCA) factor.

Logical functions are independent of any particular technology. PCA is based on the project's complexity and applies to the physical view or environment of the project.

The general approach is to identify all logical functions of the project and then:

(1) Categorize each logical function into one of five categories to determine its classification criteria.

(2) Based on the classification criteria for the category the logical function fits into, assign its classification weight.

(3) Sum all weights together to determine the unadjusted FPC.

(4) Apply a Processing Complexity Adjustment (PCA) factor to the unadjusted FPC to determine the project's total FPC.

Logical function categories include:

- External Input Type
- External Output Type
- Logical Internal File Type
- External Interface File Type
- External Inquiry Type

Each logical function is rated Simple, Average, or Complex according to classification criteria specific to the category it belongs to as shown in Figures G-1 - G-7. Each logical function is then assigned a classification weight according to its assigned rating. The classification weights are shown in Figure G-8. These figures are found in the SWAN Cost Model [IITRI 91].

Files Referenced	**Data Items Referenced**		
	1 - 4	5 - 15	> 15
0 or 1	Simple	Simple	Average
2	Simple	Average	Complex
> 2	Average	Complex	Complex

Figure G-1: External Input Type

Files Referenced	**Data Items Referenced**		
	1 - 5	6 - 19	> 19
0 or 1	Simple	Simple	Average
2 or 3	Simple	Average	Complex
> 3	Average	Complex	Complex

Figure G-2: External Output Type

Files Referenced	**Data Items Referenced**		
	1 - 19	20 - 50	> 50
1	Simple	Simple	Average
2 - 5	Simple	Average	Complex
> 5	Average	Complex	Complex

Figure G-3: Logical Internal File Type

Files Referenced	Data Items Referenced		
	1 - 19	20 - 50	> 50
1	Simple	Simple	Average
2 - 5	Simple	Average	Complex
> 5	Average	Complex	Complex

Figure G-4: External Interface File Type

Files Referenced	Data Items Referenced		
	1 - 4	5 - 15	> 15
0 or 1	Simple	Simple	Average
2 or 3	Simple	Average	Complex
> 3	Average	Complex	Complex

Figure G-5: External Inquiry Type/Input Part

Files Referenced	Data Items Referenced		
	1 - 5	6 - 19	> 19
0 or 1	Simple	Simple	Average
2 or 3	Simple	Average	Complex
> 3	Average	Complex	Complex

Figure G-6: External Inquiry Type/Output Part

Function Type	Simple	Average	Complex
External Input Type	3	4	6
External Output Type	4	5	7
Logical Internal File Type	7	10	15
Interface Files	5	7	10
External Inquiry Type	4	5	6/7

Figure G-7: Classification Weights by Category

The equations used for FPA are given below, and yield total function points for the project. The equations are:

$$\text{Total FPC} = \text{FPC} * \text{PCA}$$

where FPC is the summation of classification weights of each logical function identified for the project and PCA is the adjustment factor. PCA is always a value between .65 and 1.35 and is determined by the following equation:

$$\text{PCA} = .65 + (.01 * \text{DI})$$

where Degree of Influence (DI) is a summation of 14 General Application Characteristics with each having a value from 0-5. The DI can therefore range from 0 to 70.

The General Application Characteristics are (1) Data Communications, (2) Distributed Data or Processing, (3) Application Performance Objective, (4) Heavily-used Configuration, (5) High Transaction Rate, (6) On-line Inquiry and Data Entry, (7) End User Efficiency, (8) On-line Update, (9) Complex Processing, (10) Code Reusability, (11) Conversion and Installation Ease, (12) Operational Ease, (13) Multiple Sites, and (14) Facilitates Change.

0 - Not present, or no influence
1 - Insignificant Influence
2 - Moderate Influence
3 - Average Influence
4 - Significant Influence
5 - Strong Influence

Figure G-8: Processing Complexity Rating Scale

For example, Data Communications (General Application Characteristics 1) involves the transmission or receipt of data over communication facilities, including networks, concentrators, multiplexers, and locally connected terminals. If this has "Strong Influence" on the target project, then its rating is 5 as given in the rating scale.

To illustrate these equations, assume a project has an unadjusted FPC of 1000 and all of the 14 General Application Characteristics are rated a 5. The PCA is then .65+(.01*(70)) which yields 1.35. Its total FPC is then 1000*1.35 or 1350 points.

Size is then determined from a lines of code to function point conversion table developed by T. Capers Jones, SPQR, Inc. The list shown below contains the average number of source lines of code required to produce one function point and is categorized by programming language. Note that the average values are subject to variances in programmer efficiency, productivity levels, tools used, application size, as well as other factors. A great deal of estimating error can be introduced through the use of FPC to SLOC translations, therefore, it is extremely important to develop a historical data base which can be used to modify these average values.

Average values for the 30 most popular programming languages are:

Language	**SLOC/FP**
Basic Assembler	320
Macro Assembler	213
C	128
ALGOL	105
COBOL	105
FORTRAN	105
JOVIAL	105
Mixed Languages	105
Other Languages	105
Pascal	91
RPG	80
MODULA-2	80
PL/1	80
Ada	71
BASIC	64
FORTH	64
LISP	64
PROLOG	64
LOGO	58
English-Based Languages	53

Data Base Languages	40
Decision Support Languages	35
APL	32
Statistical Languages	32
OBJECTIVE-C	27
SMALLTALK	21
Menu-Driven Generators	16
Data Base Query Languages	13
Spreadsheet Languages	6
Graphic Icon Languages	4

The Function Point method of size estimation is supported by the International Function Point Users Group (IFPUG).